DOCTOR OF ENGINEERING (ENGD)

Performance Driven Design Systems In Practice

Joyce, Sam

Award date:
2016

Awarding institution:
University of Bath

Performance driven design systems in practice

by
Sam Conrad Joyce
MEng (Hons), University of Bath (2008)

Submitted to the Department of Architecture and Civil Engineering
in partial fulfillment of the requirements for the degree of
Engineering Doctorate in Systems
at the
University of Bath

August 2015

Author . . . .
Department of Architecture and Civil Engineering
August, 2015

Abstract

This thesis is concerned with the application of computation in the context of professional architectural practice and specifically towards defining complex buildings that are highly integrated with respect to design and engineering performance. The thesis represents applied research undertaken whilst in practice at Foster + Partners.

It reviews the current state of the art of computational design techniques to quickly but flexibly model and analyse building options. The application of parametric design tools to active design projects is discussed with respect to real examples as well as methods to then link the geometric definitions to structural engineering analysis, to provide performance data in near real time. The practical interoperability between design software and engineering tools is also examined.

The role of performance data in design decision making is analysed by comparing manual work-flows with methods assisted by computation. This extends to optimisation methods which by making use of design automation actively make design decisions to return optimised results. The challenges and drawbacks of using these methods effectively in real deign situations is discussed, especially the limitations of these methods with respect to incomplete problem definitions, and the design exploration resulting in modified performance requirements.

To counter these issues a performance driven design work flow is proposed. This is a mixed initiative whereby designer centric understanding and decisions are computer assisted. Flexible meta-design descriptions that encapsulate the variability of the design space under consideration are explored and compared with existing optimisation approaches. Computation is used to produce and visualise the performance data from these large design spaces generated by parametric design descriptions and associated engineering analysis.

Novel methods are introduced that define a design and performance space using cluster computing methods to speed up the generation of large numbers of options. The use of data visualisation is applied to design problems, showing how in real situations it can aid design orientation and decision making using the large amount of data produced. Strategies to enable these work-flows are discussed and implemented, focusing on re-appropriating existing web design paradigms using a modular approach concentrating on scalable data creation and information display.

Acknowledgements

This thesis represents work undertaken not just over a number of years, but also in partnership or collaboration with a number of companies, and institutions.

I would like firstly to thank my supervisor Chris Williams, for his invaluable guidance, especially for his pragmatic but ever philosophical support over periods of complex transition, also to Paul Shepherd who has freely given advice whenever asked. As well as the Systems Centre at the University of Bath and Bristol for creating the unique conditions in which this research was undertaken.

This research would not have been possible without the challenging and enriching environment of Foster + Partners, specifically the head of the Applied Research and Development group Francis Aish. This thanks is also extended to friends and colleges in the ARD group, Adam Davis, Martha Tsigkari, Jens Olsson, Angelos Chronis, Henrik Malm and Khaled ElAshry. Wider appreciation is given for the collaboration from many other colleagues whom I have had the pleasure to work with on numerous projects, importantly those in the engineering group Andy Coward and John Larkey, as well as members of the SMG group.

Importantly to Robert Aish for giving me the opportunity to collaborate on Design Script with Autodesk. To comrade John Harding for making this Engineering Doctorate 'journey' all the more rich. Thanks also to those in my previous position in Buro Happold where the research began. Especially the support and friendship of Al Fisher, Robert Hart and Laurent Giampellegrini, as well as the head of the SMART Group Shrikant Sharma. I would also like to acknowledge the stimulation and discussion of the wider design computation community.

Finally I would like to thank Carol See for her enduring support in all my endeavours including this thesis, and my family John Joyce and Maggie Hewlett, to whom this thesis is dedicated.

Contents

List of Figures

Chapter 1

Introduction

"Technology is the answer. But what is the question?"

Cedric Price, Symposium Lecture, 1966

1.1 An EngD

This research has been undertaken as an Engineering Doctorate. This is different from a traditional PhD and is a program devised by the British Engineering and Physical Research Council (EPSRC), with the goal of promoting cooperation and synergy between commercial and academic sectors. To this end it places researchers (in this case the author) within industrial setting but with close links to academic institutions.

This EngD was undertaken as a partnership; the primary focus being provided by architects Foster + Partners, through real projects and practice based challenges. The University of Bath Architecture and Civil Engineering department providing the academic environment and domain support, with the University of Bristol and Bath jointly providing systems and innovation management input respectively.

Figure 1-1: The people of Foster + Partners. Source Foster + Partners.

This method of study was chosen because it aligned with the authors desire to undertake research that was inherently industrial in focus and as such would have significant and immediate practical application. However equally one that is sensitive and responsive to the social and managerial aspects of technology introduction/adoption and the 'disruption' that results. Furthermore it is compatible with the belief that much of the novel applied research progress in architecture and engineering is undertaken in practice rather than in academic centres, and that there is value in exploring how it is realised in this environment.

However the drivers for this work are not solely industrial in focus; it is believed that there is academic benefit in observing the often closed field of commercial innovation, in highlighting issues that reoccur in large prominent design firms, and sharing findings on attempts to improve process from within. It is believed that this work will help others to understand the needs and challenges of innovating new technology and processes, which are relevant and effective in practice.

1.1.1 Background

The vast majority of this research has been undertaken whilst at architectural engineering and design practice Foster + Partners. The office was founded in 1967 by architect Lord Norman Foster. As a result of success from projects such as the HSBC building in Hong Kong, the Reichstag Renovation in Berlin, Wembley Stadium in London , Millau Viaduct in France and the Beijing International Airport in China, it has now grown in size and stature to become one of the largest and well known practices nationally as well as globally [Foster, 2008], [Sudjic, 2010]. The practice's early and continuing emphasis on improving the structural and environmental performance and adoption of new industrial methods have meant that its style along with other related architects have been given the name 'High-Tec' [Kron and Slesin, 1978].

At the time of writing 2014 to 2015, Foster + Partners (known as Fosters or F+P and referenced variously in the thesis as such) has grown to over 1400 people, of which more than 800 are architects. During the development of projects at the end of the last decade such as the Great Court at the British Museum (1994-2000) and Swiss Re (1997-2004), geometrically complex forms were proposed as effective solutions to enable greater performance from the building. In these cases it was to enable large single-span spaces and low wind resistance forms respectively. However the projects required external consultancy services to realise these aims. The former by Chris Williams of Bath University [Williams, 2001] working alongside engineers Buro Happold and the latter by Mark Burry supporting engineers Arup [Allinson and Thornton, 2014].

In response to this growing use of geometrical and computational complexity, the Specialist Modelling Group (SMG) was set up in 1997 as an internal consultancy to support the practice's needs [Peters and De Kestelier, 2006]. Since then geometric techniques have advanced, with specific software (often refereed to as parametric modelling) being developed by vendors specifically to support this work [Aish, 2003]. Complex geometry skills have become somewhat ubiquitous, especially among younger architects, who began eduction when these tools had matured enough to be established in the curriculum. Thus, these capabilities are now often in the design groups themselves, with advanced modelling capabilities being centred in the SMG and core skills disseminated and supported by the members.

However design technology is changing rapidly; not just in geometry definition but also integration, analytics, optimization and visualisation. There was however the view that these 'bleeding edge' techniques are often slower to be adopted into architecture despite their apparent benefits. Thus, the company's in house capability was further extended with the Applied Research and Development Group (ARD hereafter) in 2011. Its remit is to focus on advanced research required to progress the practice's capabilities both in its current projects but also in the future, not just in geometry, but also in all applications of new technology, improving the design and construction process.

In 2011, two engineering divisions where also added at Foster + Partners; Engineering 1 (E01) and Engineering 2 (E02); complete structural and environmental engineering consultants respectively, and as of the time of writing each numbers 40 and 45 people. They where established to enable the practice to better integrate design concerns for its projects, by having all of the major design stakeholders within the same company. This is a larger but similar step as previous steps to internalise groups that are typically external to an architectural practice. For example, Foster + Partners has its own model-making workshop, 3D printing facilities, visualisations and design communication group to name a few.

The author joined Foster + Partners in 2012 with the role to work as part of the ARD group but specifically to focus on proving support R+D activities between ARD and E01. This in in keeping with the authors previous experience in engineering offices primarily Buro Happold from 2009-2011, as well as the architectural modelling process as a member of Foster + Partners SMG in 2008.

1.1.2 Research Environment

The research has been undertaken whilst directly in practice, with the ARD group. Where its skills lie mostly in the ability to create custom computational process and software, the groups main outputs are twofold; focusing either on assisting and in some cases leading specific projects or automating general work-flows and design needs. Equally, the team is expected to be involved and contribute wherever relevant to a project. This is especially true of cases where our involvement saves considerable effort by those without programming skills, and also when a specific process would not be possible without a computational approach. In these cases ARD has more control and ownership over projects, which often represent the more unique and experimental projects of the office. Thus, it is often expected that more research will be undertaken in the delivery of such projects, and indeed it can be expected by clients.

The research direction of ARD, are both directed by the members of the team and the project practice requirements. Some of these are purely speculative based on the group's interests and beliefs on what would be useful going forward. However, most of the research is directly initiated from project or other team's requirements. On this experience, research is conducted and interventions are developed to solve or alleviate problems encountered during design. In some cases these issues are general; such as the inability to effectively communicate technical detail to clients. In other cases it can be very focused, often driven by functional requirements such as; a reduction in the number of unique elements in a design. In almost all cases however it involves understanding both the *hard* technical problem, but also the *soft* human definition and thus the root of the problem which comes from design decisions which preceded it.

Working Environment and its Impact on Research

The research has been undertaken with the researcher embedded in the working environment of a busy practice, as an active associate member of the company. This position has had a very real effect on the research. As an associate working full time on structural integration presents a greater reliance and responsibility to respond to design and design team requirements. Although constraining with respect to time; this also allows the opportunity and often authority to try new approaches and technologies on real projects. Conversely, being involved in the direct delivery of major projects means that production of vital design information is of greater immediate commercial concern and thus often takes precedent over more long-term research needs. Despite these issues, it has been the view of the author that the benefits of the authors situation in the company outweighs the dis-benefits due to the research being directly applied and tested in the field.

1.1.3 Previous EngD

Whilst all the new research below was carried out either at Foster + Partners or at the University of Bath by the author, it is important to note that a previous EngD project was started by the author at consulting Engineers Buro Happold, and some of the work that was published is referenced here.

This previous stand point of the work has also had a significant impact on the scope and standpoint of the research. Engineering and architecture jointly provide the same material output namely buildings however their concerns and approaches are quite different. Engineers are focused on the technical and optimal definition of both the problem and design, whereas architects are more interested in the social and aesthetic aspects. It is believed that this research benefits from the previous engineering-centric view point and that in some discussions this unique double view point is an important component to the thesis' contribution to knowledge. Indeed it is a premise in this research that greater understanding and integration of both engineering and architectural concerns can be realised through technology, and that this higher level of understanding will result in better building design.

1.2 Motivations

At a higher level this work is motivated to test the hypothesis:

Design process can be improved with respect to the functional performance of the designs derived by exploring the use of computational methods to aid design process, analysis data collection and design insight.

This was broadly a shared goal of all of the stakeholders who initiated this research project, specifically the head of Applied Research Francis Aish, academic supervisor Chris Williams and the Author. However each stakeholder had different motivations entering into the research based on diverse requirements. And it is worth identifying these separately to understand the pressure on the research goals and direction.

Commercial Motivations

The goals of Foster + Partners are very much a mixture of bottom-up emergent needs brought about by delivering projects and top-down initiatives to improve the overall structure and running of the practice. With the introduction of the engineering groups it was believed that there was a new opportunity to improve the integration of architectural and engineering concerns (a desire of the F+P group since inception), with the approach based on the increased contact and commercial openness (including liability) between the two groups. This is in contrast to the traditional interfaces which are often initially collaborative, but can become defensive and a hindrance. Arguably due to often directly antagonistic goals that both parties are expected to deliver; exciting novel designs as compared to safe well understood solutions for architects and engineers respectively. But also in part due to the relatively high levels of litigation that exists during building projects.

As a result one primary research goal was to investigate how novel methods could be tested which integrated different disciplines. By removing the commercial and legal divisions between the professions there was interest especially by the ARD group leader, but also more generally at senior partner and board level as to what could be achieved. This is of special relevance during the early concept development phase, where there is much potential to improve the design if engineers and architects work together. However, typical roles make this kind of collaboration less likely to happen.

Academic Motivations

The higher level concern of the systems centre and EngD program was to promote and realise good university and commercial interaction. As such, few concrete constraints or goals were placed on the research except to expand the academic knowledge of the design industry and problems which are relevant to academic study. As part of the joint Bath and Bristol EngD Systems Centre, there is an emphasis on the research investigating the 'soft' social and commercial aspects as well as the 'hard' technical processes and interactions that make up the researchers domain but equally this aligned relatively well with the research's goal to investigate design process.

Personal Motivations

The author's goals were to explore methodologies which enable the above goals of both F+P and the university. To see what role computation has in enabling the improvement of the design and engineering interface.

The author considered that by observing interaction between team members, that a more representative understanding of how integrated design processes worked (or failed to work) could be obtained. It was previously identified by the author that in engineering practices problems often come already defined and constrained frequently the result of a misguided (or not guided at all) design process. Typically these were very functional requirements of the design which were failing and jeopardising the success of the whole project. However it was felt that if better decisions were taken earlier on then this position could be avoided altogether. This was a hypothesis that was possible to test in the context of F+P. The unique environment of F+P offered an opportunity to observe and intervene in design activities more easily (and with less commercial risk) than the traditional design meetings between two separate companies. Furthermore any studies and experiments could be extended into multiple design sessions as the engineers were permanent members of the company with the same people consistently involved in projects. Rather than the often changing relationships and collaborations between different companies and teams of engineers and architects.

Unifying goals

The situation provides a very special opportunity for all those involved to try to work differently. Indeed in some cases this proved to be a necessity to work efficiently and deliver on projects. The office provided a platform that allowed and encouraged interaction in the main. However this also comes with some unique problems of its own: as in some cases constant interaction and changes work contrary to delivering insight and meaningful input. Another facet was the impact of the research on those working in the practice. Whilst amongst those involved there generally was a positive approach to changing the existing modes of working, respect was also required by the researcher when disrupting processes in not eroding peoples roles unjustly, such as reducing the value of their input. Explicitly to enhance collaboration, not diminish or remove peoples contribution, which in turn helps gain support for the new methods.

1.3 Research Aims

For research with such broad scope the initial aims were equally wide ranging. They centred on understanding and defining the domain of study. As well as being sensitive in that period to any recurring problems that are encountered so as to highlight areas of potential productive focus.

The selected areas of focus where investigated and better defined over periods of trial and error experimentation of potential interventions, as and when relevant projects where available to test these methods on. Thus the initial wide ranging experimentation giving way to deeper research into more specific areas. Such as the general application of computation in engineering and design, to eventually the use of automated analysis and data visualisation for design decision making.

As the process continued the aims become more focused and practical towards identifying and solving a set of interlinked issues. Resulting in the final proposals of intervention, implementation, testing and resultant conclusions.

1.3.1 Research Questions

This research aims to improve the state of integrated building design by applying better use of computational design. As such it requires a range of questions to be addressed. The research questions may be categorised as *exploration* of current practice and *interventions* in that practice.

Explorative:

- How is design undertaken in practice?
- How is engineering undertaken in practice?
- How do the disciplines currently interact in practice?
- What does technology currently do or not do to help this interaction?
- What issues arise in this process?

This research has to understand how design is undertaken in general and what are the social and technical challenges. It also asks where the most important decisions are made in that design process, who are the key stakeholders and how does their role and power change over time? How are design teams forged and how do they work together, especially the engineer architect relationship?

There are also more technological technical questions. What are the current digital work flows for teams and individuals? How is information passed between the disciplines? What are the main problem points for such work-flows? All of these require a degree of answering to help understand the environment that the research is operating in, then to pose relevant interventions.

Interventions :

- What can be done to improve the situation?

- Which ways are more effective?
- What happens when they are applied to real projects/situations?
- Are these methods actually more effective than current practice?

At the stage of interventions similar questions must also be asked of the teams but looking at the improvements (or lack thereof) relative to the existing situation. Questions orientated around the actual potential of technology to help must also be asked; e.g. Is it possible to improve how people work with technology?

1.3.2 Performance Driven Design

Whilst the research questions are wide ranging and open in nature, there is a base postulate that by introducing appropriate performance information into the design process that more effective outcomes will result. This is a strong proposition but one which is integral to the research stance taken. As this work focuses on combining different disciplines therefore more cross domain data, but also by using computation as a means to speed up or extend processes, which inevitably results in more data being produced on the design.

This research chooses to direct its technical efforts on intergeneration and generation of data which is of a more quantitative nature specifically that which relates to performance metrics of the design. These metrics may be broad and must relate to the projects at hand, however there is a belief that some generalisable patterns may emerge demonstrating more productive ways of arriving at designs which have both architectural merit whilst also efficient in an engineering sense.

In this way it differs in scope from other digital design research efforts which focus on developing purely advanced geometry without concern for pragmatic concerns. Neither does it typically take on singular performance issues to realise or improve buildings such as design rationalisation. Instead aiming to investigate the effect of informing more deeply members of a design activity about the range of important metrics so that they may make their own decisions and subjective calls but based on relevant objective clear data.

1.4 Contribution to Knowledge

It is believed that this work contributes to knowledge in several key ways. Firstly it sheds light on some of the processes and approaches that are undertaken by high end practices such as Foster + Partners. It presents some of the issues in current practice as perceived by the author especially with respect to digital work flows, and it is believed that this experience will be of use to other researchers when they are trying to formulate problems to solve. It presents some key observations on integration from the point of view of someone who has had the opportunity to operate in both fields and thus gives a unique view.

The second key contribution is the development of technological interventions in response to commercial needs and direct feedback within practice on their effectiveness. It is hoped that this body of work will enable not just the author but others to focus efforts on research that has real practical commercial impact and benefit. Especially as this applies to improving design projects irrespective of the complexity or cost by introducing computation and metrics at the early stages of decision making.

1.5 Methodology

This research does not follow the classical scientific method when it comes to methodology. The topic of research, the researchers position within the system to be examined and mode of interacting with the system would not allow the typical systematic and controlled experimental approach to be applied.

There are some existing methodologies that pertain explicitly to effective alternatives to the scientific method of researching in large social organisations such as companies. Furthermore there is a growing body of study looking into effective methods research on organisations and how to implement technological intervention and change on them as surveyed by Checkland [Checkland, 1999]. Some significant methodologies developed for the engineering and management domains include soft systems methodology [Checkland and Scholes, 1990], applied systems thinking [Flood, 1993], systems methodology for management [Jackson, 1991], the 'Fifth Discipline' [Senge, 2014]. These are all of relevance as the main thrust of the research is concerned with identifying problematic areas within the design process and improving this via technology. They also identify the complexity inherent in the action research position, where the researcher is a participating member of the system that they are trying to change. Being able to do effective research within a constantly evolving situation is of critical importance, as this is the prevalent mode of intervention available to the researcher. As such these methodologies have been important in the development of the methodology for this research.

However with respect to direct adoption of these methodologies there are some issues. Many of the aforementioned approaches are either too general or socially focused to be relevant in a design technology context, which is the case with the management literature. Or alternatively orientated towards more pure engineering domains such as aerospace or mechanical engineering, where engineering is the prevalent activity and activity is much more systematised and process driven.

This is not consistent with the problems encountered in building design integration where the aesthetics or architecture of a design is of equal importance to the functional engineering requirements. The process is also often led not by engineering logic but by the architect following architectural considerations. Where the interchange between engineering and architectural considerations results in unique processes being developed per project. Furthermore design being a deeply technical activity has less to benefit from a solely managerial overview of a problem.

As such the issues investigated by the research are quite specific to disciplines where there is a synthesis of engineering and design in equal measure and this requires a different methodology. Some researchers have highlighted these issues, most notably Rittel and Webber who characterise these design tasks as 'wicked' problems [Rittel and Webber, 1973]. Quoting directly wicked problems as defined by Rittel and Webber are claimed to have these features:

- There is no definitive formulation of a wicked problem
- Wicked problems have no stopping rule
- Solutions to wicked problems are not true-or-false, but good-or-bad
- There is no immediate and no ultimate test of a solution to a wicked problem
- Every solution to a wicked problem is a "one-shot operation"; because there is no opportunity to learn by trial-and-error, every attempt counts significantly
- Wicked problems do not have an enumerable (or an exhaustively describable) set of potential solutions, nor is there a well-described set of permissible operations that may be incorporated into the plan
- Every wicked problem is essentially unique

- Every wicked problem can be considered to be a symptom of another problem

This helps to identify the issues present when attempting to research in a design context. This raises at a philosophical level questions about rationality as applied by designers. As Coyne highlights "rationality has already been framed according to various agendas" [Coyne, 2005]. Coyne also rightly discusses the limitation in solutions that defined approaches or 'Tamed problems' can provide. Whist there are examples of methodologies developed to tackle design centric problems, most notably by Alexander [Alexander, 1964] Asimow [Asimow, 1962] and Jones [Jones, 1992]. These present methods of design and address the issues of how to systematise this process, but they do not look at the application of a disruptive technology (computation) to this already complex interaction. There is a growing body of research which has taken place at the nexus of design, engineering and technology, which is uniquely coming out of architecture and design itself [Coenders, 2012a], [Derix et al., 2011]. These all present their own custom approach derived from existing methods but importantly adaptable to the changing context of the research. This work looks to these applied methods more than the existing traditional approaches, and as such implements its own methodology, responsive to the context and problems at hand.

1.5.1 Research Methodology

The first phase of the research involves understanding the human interaction present in the field of design. Whilst there is some technology used, this is predominately a soft-system where the understanding of the interaction between team members is key. As such, data triangulation has been employed as much as possible, especially when making conclusions and proposing directions for intervention. This involves finding multiple case studies, including mixing secondary data sources from other researchers in practice. In this way it is hoped the directions derived

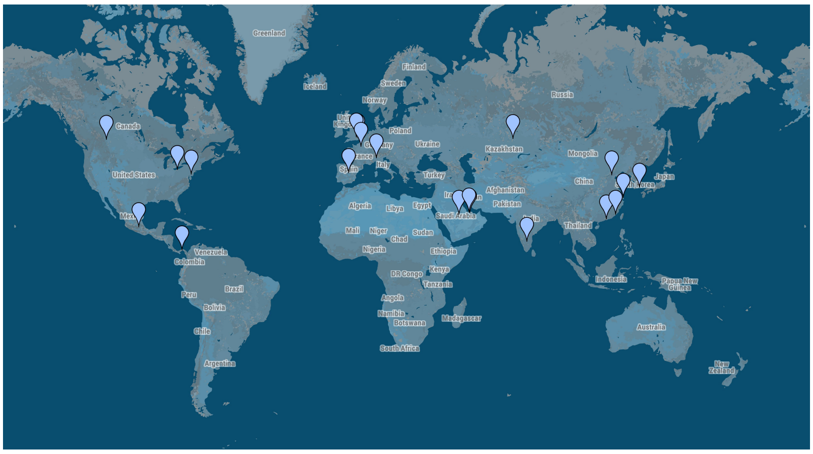


Figure 1-2: The projects undertaken by the author in this work are distributed throughout the globe.

will have maximum relevance not just in Foster + Partners but elsewhere also.

1.5.2 Intervention/Development Methodology

The major novel work of this research comprises of direct intervention into live projects with new technological approaches. As such this has two components the development of such interventions technologically, and then critical assessment of the application of them into practice. This splits the work cleanly into 'soft' research providing context and inputs for 'hard' technological research, both of which require different methodologies.

Technological Research

This research forms the *hard* technical 'solutions' to the *soft* system problems previously highlighted by the ethnographic type research. This work in many respects has easier quantitative properties with which to compare with existing methods. Such as measurements of or improvements in time taken to build a structural model for example.

The technical research will be used to define and develop the tools and techniques, mostly software and computational systems to then intervene in design activities. In this way much of this work will focus on translating existing research and technology to be more relevant to the problems highlighted by the 'soft' research.

Here a conventional research methodology can be undertaken with literature reviews of current research into design based problem solving in building design but also different industries. Implementing experimental technical solutions which are a synthesis of the existing solutions tailored to the general design issues identified over the course of the work and the specific design tasks at hand.

It is intended that any solutions will be built up in a iterative fashion and further developed or ceased based on their perceived contribution to the wider 'soft' issues.

Action Research

The key use for the technological research is to improve the integration and ultimately productivity of design teams however the measurement of the effectiveness of this is non-trivial. Here, improvements can be categorised into those affecting process and or the actual design output. For a measure of effectiveness to be derived in any pragmatic sense, the methods need to be tested in real situations. As such action research will be employed in practice.

This involves being actively engaged in projects as an equally involved and invested member of the design team. At the same time the researcher must be critical of said design activities firstly to gain insight into how the social processes occur and analysing where there is room for improvement. From this basis interventions can be proposed based on prior more research from the same or different fields and implemented on the live projects with the outcomes monitored. With the improvements, if any, derived from comparisons with previous case study examples, and cross comparison of projects undertaken before and after the research interventions. Some of this is measurable especially with respect to the quantitative improvements in a design performance (reduction in steel tonnage or running time for example). However much of the improvements will be qualitative and so methods to capture this, including reflective reviews of case studies will be employed to assess and understand the impact of the technological interventions.

The ultimate aim is that a level of inductive understanding of and improvement to the activities researched will be built up. A central problem in proving objective improvements is the inability to 'repeat the experiment' to see if a technology augmented solution did in fact have an effect over a 'normal' process. Neither is it practical to have different designers work separately on the same scheme just to find out if using a different process is indeed more effective. Instead this work relies on gaining qualitative feedback from a range of participants frequently relying on their and the authors comparison with previous processes.

Despite this, the benefit to undertaking a action based research approach in a commercial professional context as large as Foster + Partners is that similar problems and design scenarios occur relatively frequently. This paired with the capabilities of the author and authors group which attracts involvement in certain types of designs, may mean that the design process is more similar and thus more comparable than in many practices.

Where possible feedback is obtained both formally and informally. Primary feedback is obtained by reflection on the past activities allowing for criticism and insight. At certain points formal data collection about the research will be required to capture in depth the options of others via interviews. These interviews allow for in depth directed questioning, and is able to compare a number of participants which is of value with the range of roles in a typical building design process. Frequent interviews are not practical in a working environment especially around live time sensitive projects. It is also a concern that taking too much time over this will actually harm the research as it will be viewed as too much of an encumberment to design teams and thus not adopted. As such feedback capture though informal routes such as personal discussions and comments will also be used and seen as valuable. In collecting this data one much be aware of the implicit bias of the researcher doing the questioning on something that they are invested in, however in this case it is unavoidable due to the level of domain knowledge required to meaningfully inquire others on this topic.

1.6 Scope of the Research

Owing to the wide ranging domains that are brought together here namely; integrated-design and computer-aided-design some sensible boundaries must applied to the research. Whilst the research is concerned holistically with the integration of most key stakeholders in building projects, the focus of the research is primarily on architectural and structural engineering integration. This is in part due to the bias of the author's background experience in structural engineering, and also because there is benefit in looking in more detail at the specifics of integrating two concrete disciplines rather than just general approaches. Especially those whose concerns are both quantitatively and qualitatively different but closely coupled.

Furthermore, this research takes as an axiom that technology can act as an aid to existing processes. As such, whilst there is a reasonable level of analysis of projects and project teams, emphasis is placed on the technological processes and peoples use of technology, as these are the processes that will later be modified by further interventions.

1.7 Research Philosophy

The philosophy of this research is that by correctly employing technology good design decisions can be more easily taken. It accepts that often applying these methods further complicates the design process especially initially. However it is also believed that the considered strategic application of new methods early on can pay dividends as the design progresses. To these ends, it is very much the authors position that effective interventions do not necessarily have to come from complex methods, and that relatively simple application of technology and changes to process can also have significant impact. However, for these methods to be adopted in a commercial setting they must be shown to have a positive contribution during all stages, otherwise there will be understandable resistance to their use. Thus, this research looks to methods that are incrementally advantageous but with the goal that they should drive long term change. This is important to fulfil both commercial and academic goals. In may ways this development strategy is analogous to the evolution of new capabilities in organisms. Over the iterations continuous benefit must exist but from this significant advancements and paradigm shifts can be achieved as is the case with eye development.

1.7.1 Wider Trends

It is an integral part of the philosophy that there is much relevant research already present in other fields which could have a significant relevance and impact on design. Although much of this is not directly associated with the building design industry; such as the internet, economics and computer programming, this is a view that is consistent with other researchers in the field [Coenders, 2012b].

Owing to the social and process orientation of this work, it is beneficial to observe other productive areas which have had more research attention due to their large market share such as social media technologies. The research hopes to iden-

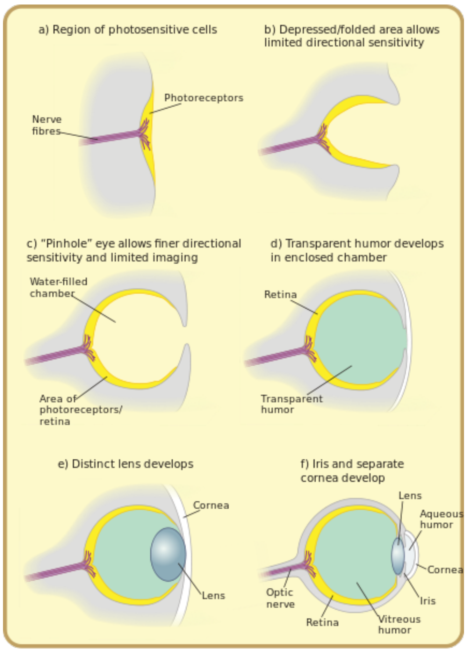
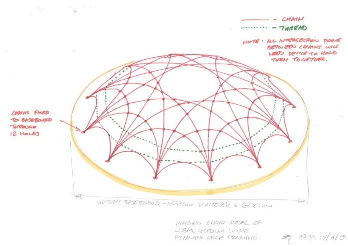
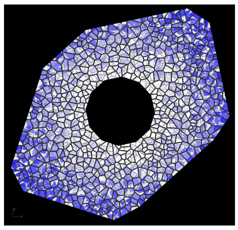
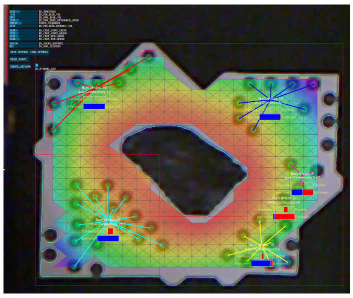
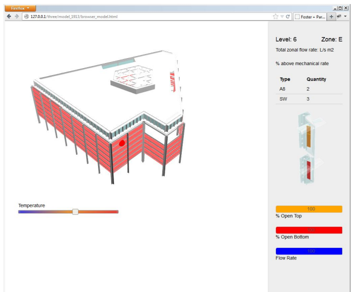


Figure 1-3: Schematic diagram of the progression of the evolution of the eye, showing continuous competitive improvement at each stage and thus higher chance of system continuation. From [Matticus, 2006].

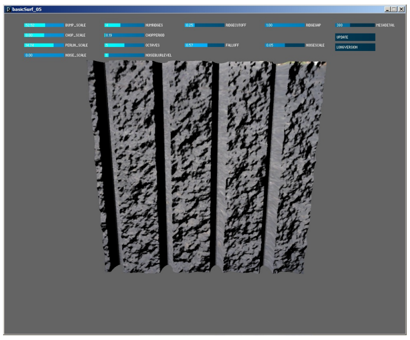
tify successful methods applied and consider their suitability in the design field.

1.8 Case Study Outline

As described in the methodology section, this research has been undertaken whilst actively involved in live projects. This has significantly shaped the content and direction of research. As such, a list of selected projects the researcher was involved with is shown by way of conveying the range of engagements that are included over the duration of the thesis. Of which an expanded version of the case studies overview can be found in Appendix B. For any more in depth detail the reader is invited to refer there.

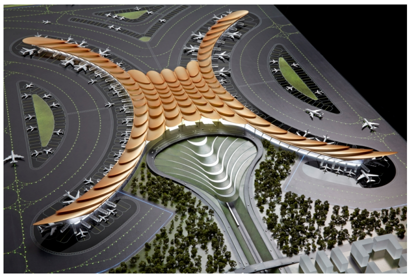
Project Description	Image
World Cup Main Stadium Qatar Study to create hanging chain model for the large aching roof geometry of the Qatar World Cup central stadium. Involving digital modelling of the chains, to determine the physical chain lengths.	
SAMBA Tower Rug Design Parametric modelling undertaken to model a geometric pattern for main boardroom space of SAMBA Bank. Geometry was visualised using Augmented Reality for client feedback.	
Work Space Tool Interactive design tool for measuring social quality of an office configuration. Included an interface to allow direct manipulation of a physical office model, which via computer vision updated the digital configuration and recalculated design metrics.	
Bloomberg Development of interactive web interface explaining the natural ventilation strategy for the new Bloomberg office in London. Interface demonstrated with a 3D model in the browser the configuration of vents at a macro an component level.	

Shanghai Bund



Investigation of computer generation of rusticated stone façades for uban development in Shanghai in conjunction with Tomas Heatherwick. Resulted in specialised stone designing tool to produce unique 3D and texture mapping output for design development and visualisations.

Beijing South Airport



Parametric definition of a highly sculptural 'Phoenix' like roof and support structure for a competition entry. ARD group and the author were main coordinators of the geometry with all parties from visuals, to integrated engineering proposal including space frame structural definition.

Thames Hub

Global econometric analysis exploring commercial viability for a larger hub airport in London. Web visualisations of complex data both geographical, political and economic to develop a narrative for the creation of a new Hub Airport Proposal.

Bangalore Residential

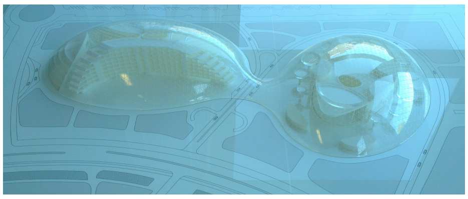
Integrated design project using a voxel based volume to seek a configuration for optimal airflow, views and insolation of a residential development. Structural using genetic algorithms optimisation was undertaken to develop a effective support strategy with the binary state of the walls.

Madrid Stadium Roof

Development of lightweight roof to increase the covered area of existing stadium. Creation of fully parametric model of various options, as well as integration to analysis software for structural appraisal. Further development of light-weight tensile options of roofs with pre-stressed cables.

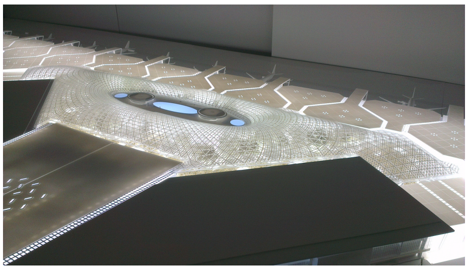
Astana Expo

Fast track development of cold weather bio-dome like containment structures for a proposed hotel and exhibition hall. Dynamic relaxation applied to create minimal grid shells. Then integration with analysis software for basic structural appraisal and buckling analysis.



Doha Airport

Generation of atrium space using a roof geometry inspired by the Great Court roof. Dynamic relaxation was applied to generate maxim shell action in the structure, with integrated analysis to test its effectiveness. Additional studies developed a fractal shading solution to minimise solar gain.



Cleveland Clinic Roof

Development of early stage atrium roof geometry. Parametric definition used to enable design exploration, with integrated structural analysis. Leading to large scale generation of design options with performance data visualisation, to understand engineering and architecture trade-offs.

Mexico Airport

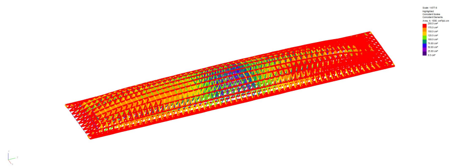
Development of main integrated roof/column continuous enclosure, including the winning competition entry and scheme design. Form created using dynamic relaxation principals, with sophisticated initial griddling to achieve an elegant but structurally efficient result. Collaboration with engineers and fabricators to optimise the potential build-ability of the form.

UAE Expo Pavilion 2015

Initial definition of complex sculptural GRC panels, to evoke sand dune geometry. Leading to detailed construction optimisation on the panels, to maximise the natural look whilst minimising cost complexity and construction time. Large parallel 3D model creation process, developed to produce fabrication documentation.

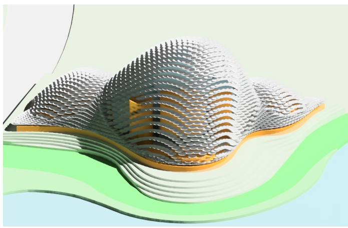
Stadium in Paris

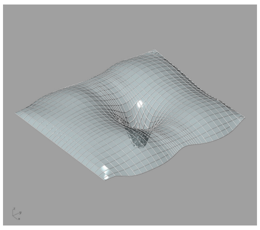
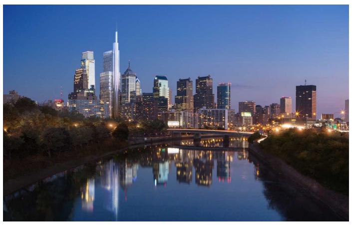
Development of large beam like movable roof with ulta-thin element length. Parametric definition of space fame with optimised section thickness, to minimise deflection.



Busan Opera House

Geometric program definition of pod like forms for containment of opera house competition. Integrated façade with decorative but functional water capture devices prosed and modelled.



Xiamen Cruise Terminal Exploration of form found integrated column and roof element as interior focal point of cruise terminal. Use of inflated dynamic relaxation for a efficient yet dramatic grid.	
Project Liberty Design investigation into roof garden's protected by geodesic dome like forms. Development of basic geometry for visual and engineering consideration via a parametric model.	
Tocumen Airport Study of structural modal design of airport link bridge. Analysis of whole viable design space undertaken to ascertain effective solutions. Web visualisation used to explore relationships in the input parameters and the complex modal frequency responses.	

These projects have been detailed in this thesis as they allowed for investigation into the application of computation and performance metrics to be used in live design scenarios. This is not an exhaustive list of project undertaken by the author at Foster + Partners but gives an impression of the range of work.

The authors approach for engagement in projects was a mixture of focused targeting, as well as those which came to the group in general and were covered by the author owing to relevant expertise. The goal was to find case studies which had a mixture of stakeholders in both engineering and architecture and more generally those interesting trade-off between different performance metrics and aesthetics.

Typically this attracted projects which where sufficiently complex to benefit from this computational, over better understood design problems.

1.9 Chapter Outline

The chapters outlined below attempt to encompass the main fields of productive study that the work has realised. Broadly speaking, the topics and thus chapters are progressive as was the development of the topics. However, following the methodology of the work, different topics and key ideas were progressed partly in parallel over the duration of the research. This is in keeping with the emergence of live projects which called for the application of different methods developed. It is a testament to the usefulness of the work that these developed capabilities have been commercially relied on more and more throughout the research period and to this day.

The chapters delineate the significant periods of progression of the research. Each has its own integrated literature reviews and secondary research to introduce the area of study and or development. With the progression of chapters mapping the development of the novel research rather than the exact temporal series of projects. It is the view of the author that this aids in understanding of the evolution of the research. As some projects relied on methods developed earlier on by the research, without any extension to them, and as such are discussed (if at all) in the chapter describing that specific advancement rather than later.

What now follows is a brief description of the chapters.

Chapter 2 introduces the key modelling technologies that have lead to widespread use of computational design in architecture, as well as explaining efforts to link this to engineering analysis.

Chapter 3 details efforts to optimise designs relying on computational design to drive the design representation and introducing algorithmic methods to improve the performance metrics.

Chapter 4 presents research and investigations into methods used to explore visualise and understand the design spaces created by computational design methods.

Chapter 5 presents reflection on these interventions leading to developments where the previous methods have been consolidated both technologically and in projects.

Finally chapter 6 discusses and brings to a conclusion what has been covered as well as future directions of this work.

Chapter 2

Representation and Evaluation

> "The problems we face cannot be solved at the same level of thinking we were at when we created them"
>
> Albert Einstein, 1947

Design is an activity that encompasses the definition, problem solving and communication for production of an artefact. Distinct from craftsmanship, it is an activity separate from the construction or manufacturing of the artefact. Nowhere is this more true than in architecture, where owing to the scale and unique nature of each building, designers are required to work separately from the construction process. Requiring planning and anticipation to mitigate potential problems often years before the actual creation of any building or mock up even. However naturally this is preferable to a trial an error approach, which at the level of a large building or infrastructure project would be expensive in the extreme if not virtually impossible. Thus designers rely on experience to pre-empt requirements and solve issues before they arise. This is supported by in-depth modelling to generate and visually or analytically test effective designs, as arguably only by externalising their intentions via mediums such as models and drawings can they be collaboratively and meaningfully discussed. Due to these demands communication is an

Figure 2-1: 30 St Mary's Axe; Internal concept sketch, client visualisation, detail documentation and construction. Source Foster + Partners.

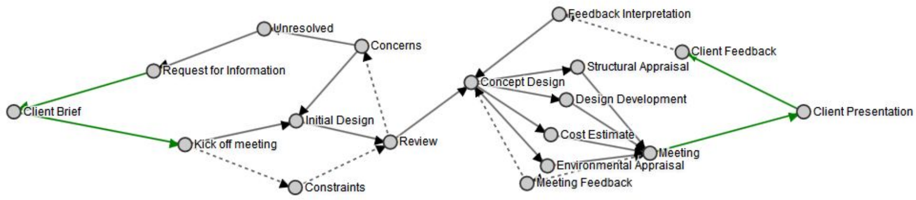


Figure 2-2: Visualisation of information flow between different processes in a concept design phase

important part of their role and this is principally directed in three directions:

1. Towards clients:- ensuring that proposals are fulfilling their requirements.

2. Towards contractors:- directing them to be able to build the proposals.

3. Towards themselves and other designers:- to enable exchange of information on proposals for evaluation and improvement.

This production and exchange of information is complex; it is iterative, concurrent and contains numerous feedback loops based on internal and external appraisal and engineering analysis. An example communication diagram that attempts to map the typical information flow of the early stage of a project is shown in figure 2-2.

This research focuses on design as a process of modelling analysis communication, investigating how computation has changed this process, and how it could further change it positively in the future. By looking at architecture and construction we examine an environment which is rich in interactions between professions with differing points of view, and thus see how different stakeholders of a design and the design-process can be integrated.

2.1 Representation and the Rise of Computer Aided Design

Computers have been proposed for use in architecture since at least 1962 with the work of Ivan Sutherland [Sutherland, 1964]. Whilst his work was revolutionary and said by many to be the "birth of computer graphics", these methods were not widely adopted within the industry until much later. Mainstream CAD packages did not emerge until over a decade after, with Intergraph IGDS in 1974, Autodesk AutoCAD in 1982 and Bentley Systems Microstation in 1985. Now, however computers are ubiquitous and central to design, especially for production of documentation such as drawings and renders. With the exception of a few mostly small firms relying on hand drafting techniques. Furthermore many government procurement programmes are specifying strict requirements for digital document delivery. For example the U.K. Government Construction Strategy only accept prospective designers who comply with building information model (BIM) standards for documentation and tender from 2016 [UK Govnemnet, 2011].

2.1.1 Development of CAD, The Automation of Representation

The paradigm of Computer-Aided-Design (CAD) emerged from the work of 'Sketch Pad' by Sutherland around 1962 [Sutherland, 1964]. His system was unique at the time by presenting a way to view and interact with the computer. Displaying 2D plans and 3D visualisations on screen and creation and manipulation of geometric objects via a light pen. This represented the the first digital method of producing design drawings. Borrowing computational concepts applied effectively to engineering problems but with a natural drawing like interface and ease of use. It also incorporated more sophisticated features, taken from constraint programming and dependency based behaviour. In many ways this was ahead of the commercial CAD environments that emerged in the 80's and came into wide use. Programs

Figure 2-3: The development of the F+P office through 1985-2014. Drafting boards have progressively made way for computers. Source Foster + Partners.

such as Bentley's Micro Station which Foster + Partners still standardise on or Autodesk AutoCAD [McCarthy, 1990]. This first generation of widely adopted CAD packages, were adopted arguably because they faithfully reproduced many features of the drafting table as a way of lowering the barriers to entry to new users by relying on a paradigm they understood. A prime example being digitisers, able to computationally store and reproduce drawings. These systems where adopted due to the advantages over traditional analogue methods, most obviously the ability to reproduce, amend and reprint drawings but also qualities such as improved line and text accuracy, auto-hatching, reduced physical storage size and actions such as 'undo' and 'zoom'.

It is however noted that these early systems where sufficiently close to existing drafting approaches as not to be alien to users. After a sufficient period of adjustment from hand drawings to CAD, 3D become more commonly accepted. As well as the aforementioned major players new systems where developed which focused on 3D, such as Rhino 3D in 1998. Whilst the design and visualisation of buildings in 3D space was not possible on paper, the creation of forms in this system was analogous to drawing but with space lines, and planes replacing lines and hatching respectively.

Programs such as Google's 'Sketchup' deviates from this constructive outline paradigm of line and edge based manipulation, for example the ability to 'pull' and 'push' to create solid surfaces which whilst being intuitive has no natural real world parallel [Murdock, 2009].

2.1.2 DDD, Constraint Based Systems

In parallel to these systems and often as add-ons to them, were emerging formal methods to represent and manipulate more sophisticated geometry. The most well known being perhaps the B-Spline cure. This was a geometric definition which

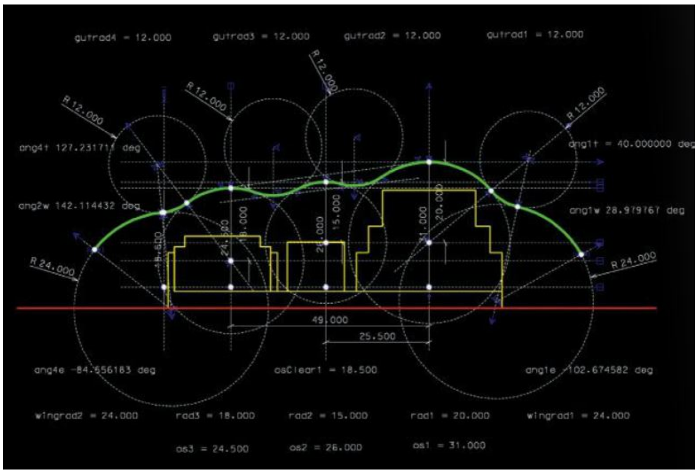


Figure 2-4: Dimension driven design model of the Sage Centre Gateshead. Source Foster + Partners.

although mimicking the behaviour of a spline (a wooden or metal strip used to make flowing curves), introduced a level of indirect definition to geometry.

Control of complex forms was further expanded by development of tools such as Bentley's 'Dimension Driven Design' (DDD). This controlled geometry by allowing users to define relationships and important input values. Then, if possible, these computational constraints were solved and the satisfying geometry drawn. This is an important step in that it abstracted the user away from direct manipulation towards one which relied on computational approaches to solve the design issue. These methods had proven to be very effective and have been successfully applied for many projects especially at F+P, on buildings such as the SSE Hydro Area Glasgow [Bentley, 2014] Swiss RE London, Beijing Airport Terminal 3, The Sage Gateshead and Greater London Assembly Building [Peters and Peters, 2013].

2.1.3 Building Information Modelling

A recent trend in representation is Building Information Modelling (BIM). The use of BIM has been somewhat ambiguous, being interpreted as both a data-centric managerial approach to design as well as the use of BIM software such as Autdesk

Revit, AchiCAD or Bently ECOSim.

BIM as a technology directly borrows from the idea of object orientated programming, where the model is made of a database of elements, each with their own properties. These elements hold relationships to one another, for example determining a wall to be spanning between the ground and first floor. Each element being an instantiation of a base class or abstract object, with the objects defined by the software or product vendors. As BIM enables the representation not just of graphical objects (eg lines) but also complete construction elements referred to as 'families' made of up of components, all of which are controlled by the super-component (such as a wall definition with a layered build up).

This has many benefits over existing definitions such as classic CAD geometric constructs, as the BIM element can be interrogated for properties as well as modified either directly by the user or via associated elements. This can be used for clash detection, quantity take-off for cost estimations, as well as material properties for structural or environmental analysis work flows. This method has proven to have significant benefits in drawing extraction where the object based model can then be reinterpreted for the derivation of specific sections and elevations with appropriate build ups, without having to generate those separately. As such it is becoming a expected and required standard for design documentation delivery.

However these systems do not address issues the logic of *how* or *why* to place those objects. Whilst there is some generative intelligence by linking objects via their properties, there is rarely a high level of sophistication in these relationships. In most regards, BIM is still very much tied to the paradigm of manual model creation. Thus, whilst the method of representation is improved and deepened, the generation and re-iteration of the model is still labour intensive. Furthermore there is significant literature documenting the issues of having too much information during the creation period, and this slowing down model generation [Pena De Leon, 2014], [Holzer, 2011]. The author would also add that for data as

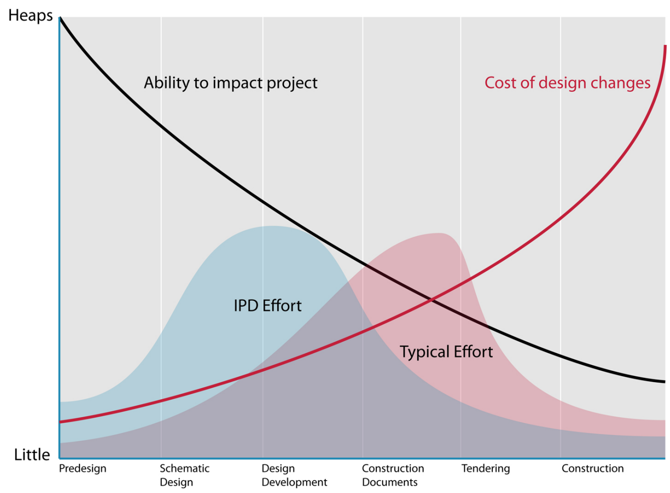


Figure 2-5: The MacLeamy Curve, as represented by Davis [Davis, 2013]

well as too *much* it, it is often of the wrong *type*. For example overly specific information such as window or door schedules is abundant in BIM. When at concept stage more general 'ball-park' type data is of better use when making the broad initial decisions (GFA, gross insolation, setback infringement by volume). There is also the issue of the rigidity of BIM object representation stifling creativity, by requiring high levels of detail and resolution, whilst much about the design is still unknown. Conversely, this has been highlighted as a benefit by some such as Patrick MacLeamy chairman and CEO of architects SOM [1] . He takes the view that to make the best use of these tools, more focus is needed to be placed on early stage design, with better understanding as to the implications of early decisions which have the largest impact on the overall cost. These ideas are encapsulated in what has come to be referred to as the MacLeamy Curve shown in figure 2-5

Some software companies have responded to these developing requirements for concept stage BIM. One example is Autodesk's project Vasari, which focuses on supporting basic building volume design and manipulation via the 'conceptual mass' geometry kernel developed for Revit [Stine, 2013]. It provides quick and easy analysis on issues such as solar exposure and wind-flow via basic links to

[1] Talk can be seen at http://www.hok.com/thought-leadership/patrick-macleamy-on-the-future-of-the-building-industry.

analysis engines Ecotect and Virtual Wind Tunnel. These volumes can then be imported into Revit for more comprehensive design development. This offers more intuitive ways to model volumes and obtain performance metrics using predetermined output visualisations. However by doing so it imposes a relatively specific use paradigm of direct volumetric manipulation, as well as very constrained basic analysis data.

It is also worth noting that BIM software has also introduced methods for group model authoring and sharing. These use model repositories usually hosed centrally by the designers or in the 'cloud'. This allows team members to create working copies of whole/parts of existing models, make new additions dependenat on current geometry and then update their contributions to the master model, assuming it is accepted by the BIM model manager. This shows interesting distributed methods of working and developing associative models in complex situations. This is consistent with managed software code revision management systems, such as Apache SVN or GIT, which are popular tools in the computer development field. However as described by Adamu, Emmit and Soetanto [Adamu et al., 2015] there are limitations to such systems. Both in terms of forcing specific collaborative work flows and hierarchies which may not be advantageous, but also in the social communication or more specifically the situational awareness of people working remotely in this way. Interestingly systems such as Github, which is a popular provider of GIT in the cloud present a more social contexts for people to work which has been very popular and productive.

2.1.4 Conclusions

Whilst it has been shown that current CAD and BIM technology has developed towards a more detailed resolution, it has also highlighted that these methods are still tied to direct manual approaches to object definition in model creation. In part, this is due to the ease in which this paradigm is understood by being analogous to drawing or physical model making. Despite this digital infrastructure offering much improvement for representation, it does not offer much practically to help in the problem solving aspect of design.

2.2 Design Automation

As well as development in the production of design representation/documentation. There has also been progress in approaches to assisting the design process, which represents a significant break from convention in creating design representation.

With the advent of the computer in design, software companies and a growing number of advanced users are reassessing how design as a *process* of dependant decisions and definitions is undertaken. Looking to see how some of this work-load can be shared with the computer; taking the logical decisions and calculations that are required for architectural design and integrating them into generative models that can respond and adapt to changes based on these relationships. Potentially removing the need to manually recalculate whenever a design change is needed.

This is a relatively new movement, which has been rather technical in nature and has only seen larger popularisation and acceptance by the design community at the start of this decade. To some this may seem to be a natural step when using a computer to represent a design, and also to harness the logic and processing of the computer to support design tasks. However its slow rate of early adoption at least shows a different trend. Indeed the foreign nature of model creation in this way as compared to drafting techniques puts it at odds with much of the traditional architectural pedagogy exuding many from its use.

Since then much effort has been taken by software houses and active groups of users to support people who are not professional computer scientists to employ these methods [2] . As a result, there is an increase application of computational methods on real projects and in design practice [Peters, 2013].

What will follow is an introduction to the key innovations and aspects in this field, especially with respect to how this technology is applied commercially including Foster + Partners.

2.2.1 Scripting

Scripting is the exposure of parts of a CAD system by the vendors, to algorithmic manipulation by a programmed set of computer instructions; The first widely adopted user orientated scripting environment being Autolisp released for Autocad in 1986. This enables users to write scripts which are able to automatically generate geometry and automate time consuming tasks within the CAD environment. Bentley integrated Microsoft's Visual Basic (VBA) for the Microstation application, providing a full Integrated-Development-Environment (IDE) for Microstation from 2001. This exposed the base functionality of the CAD software primarily to enable developers extend the application with plug-ins. However this was also employed directly by designers with programming skills as a means to generate complex geometry.

This was employed in Fosters as early as 2001 where the methods where employed to make the London City Hall (GLA) building. Some of these tools features were also generalised and turned into 'buttons' enabling non-programmers to employ the sophisticated geometric logic without having to possess any programming knowledge. This was further developed into menus which allow the lay-user to generate for example; a diagrid structure or stair core, by simply defining some input values and initialisation geometry, such as the Foster Tool Kit (FTK)

[2] Key examples being the Smart Geometry Group, initially heavily sponsored by Bentley which has run large annual workshops internationality Since 2007, promoting the use of parametric design methods. As well as special schools that focus on the application of such technology such as the Institute for Advanced Architecture of Catalonia.

which is a company wide extension to Microstation available to all employees and supported by both ARD and SMG groups.

Code based geometry creation requires a relatively high level of technical programming skill. There is also the danger of introducing too much complexity into the design process which takes time to create and is difficult to change. One example is the Smithsonian Roof where over 5000 lines of code were written specifically to derive the geometry [Peters, 2007]. However, it is interesting to note that the British Museum roof definition was only 1835 lines [3] , which despite being written significantly earlier; and including its own non-linear structural analysis and custom 3D viewer in the same program. Whilst lines of code is probably not a fair metric for efficiency or compactness, it demonstrates the lengths required to create a whole design in this way using programming alone. And the problems in code comprehension if the code needs changing or reusing for another purpose. As the author has found out when having to adapt both the Smithsonian and Great Court Roof code for exploitation of similar ideas in other projects.

2.2.2 Parametric Design

With the introduction of 'Generative Components' (GC) in 2003, knwon intiallay as 'Custom Obbjects' [Aish, 2000] a significant attempt was made to provide a more inviting interface to programming for architectural CAD users, this was introduced more widely in the Smart Geometry Conference 2003 in Cambridge. The main innovation was to provide a more intuitive way of creating and generating geometry computationally and the logic systems they rely on. This was realised by defining a system where new geometry was dependent on existing geometry and numerical inputs [Aish, 2003]. For example a line could be produced by referencing two points, or a point could be produced by referencing a line and the length

[3] This was derived from Chris Williams' website http://staff.bath.ac.uk/abscjkw/BritishMuseum (accessed October 2014)

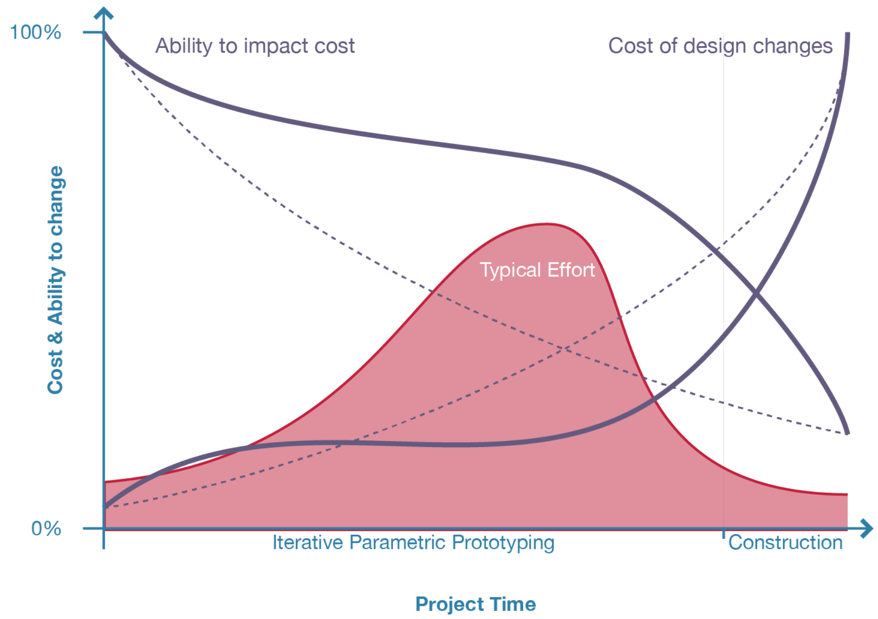


Figure 2-6: The Davis Curve a modification of the MacLeamy curve, showing the effect of parametric modelling on design change costs, as represented by Davis [Davis, 2013]

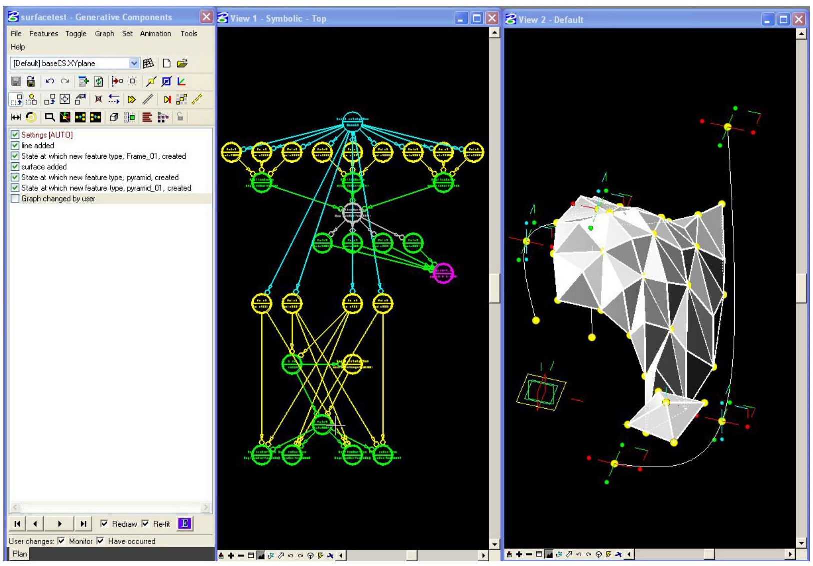


Figure 2-7: Example Generative Components design session, showing from left to right: user generated design steps called 'transactions' , the associative relationships between elements or 'symbolic view' and the geometric model view. Source Bentley Systems

along that curve to place the line. At a higher level of abstraction each element in the system can be mapped as a node with edges connecting to existing geometry or values. A complete model can be represented by a tree or more specifically a directed acyclic graph (DAG), this was visualised in GC as the 'symbolic view' as shown in figure 2-7. This graph enables the system to identify which elements are effected when element or value is changed, and update these also. This is the paradigm which defines 'Parametric Design' so called because interaction with such systems is concentrated to changing values or parameters each time creating an updated design. And these concepts have been re-implemented by David Rutten in Grasshopper for Rhino 3D [Rutten, 2012b] and has become widely adopted, with over 33.5 thousand members of the user group as of January 2015. This has also lead to the identification of a style 'Parametricism', with which is associated with architects such as Zaha Hadid Architects [Schumacher, 2009]. However the use of the work 'parametric' in an architectural context was used much earlier by Luigi Moretti [Moretti et al., 2002].

The important feature of computation is the restive ease with which many are now able to produce computational models, and explore the parameters they build into them. Other architectural firms have identified the commercial benefit of parametric tools to enable late stage changes, as one is able to change parameters or even pieces of logic and have a model regenerate with little effort. Putting these tools in the centre of a technology strategy as enablers of faster and more responsive design [Burger, 2008], [Shepherd et al., 2011]. These benefits have been discussed at length by Davis [Davis, 2013], represented by his modification to the MacLeamy curve and shown in figure 2-6. There has been concern by the same author and others on the a lack of rigour in defining these models. With the main criticism being that models which have to many inter-dependences, become unwieldy both to the user to modify quickly in response to design changes, but also for others to understand, thus creating what as been referred to as 'spaghetti' code [Davis et al., 2011]. This leads to models being inflexible to change, especially if

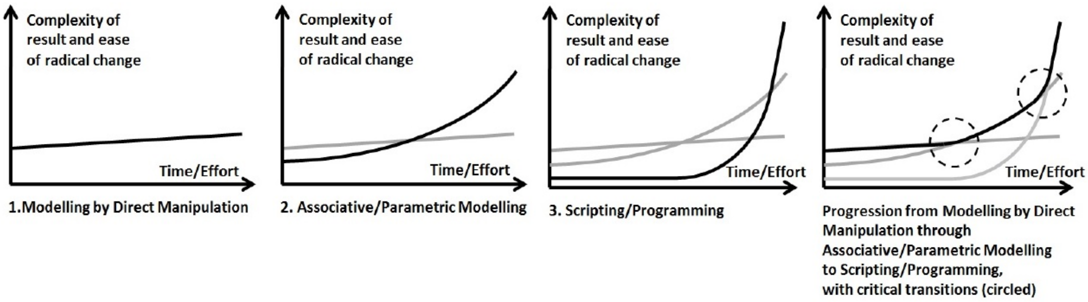


Figure 2-8: Analysis of learning curves of different environments including design script from Aish [Aish, 2012]

that change is not anticipated and therefore, not developed into the parametric model to be a degree of freedom. Leading to counteraction of the central tenant that parametric design speeds up creation of design options.

Whilst this has been the experience of the author it is believed that this approach is still preferable over manual methods, especially when required to investigate many similar options and variations.

2.3 New Developments in Computational Design

One of the major issues with both scripting and graphical programming has been the high investment in time required due to the complexity of these systems. This pedagogical problem is one that has been identified by academics and practitioners alike [Aish, 2012]. However new approaches to parametric design have been developed by Aish that demonstrate new programming paradigms specifically for designers resulting in a new domain specific language 'Design Script'. This research which the author was an early contributor to, attempts to allow a more high-level and compact definition of logic to help users read and write scripts. Aspects of this language mimic a recent wave of computer languages such as Ruby and Python which diverge from older programming languages like C++, these aim to operate at a higher level of abstraction from basic operations, saving programmers from resolving time consuming issues like object-creation indexing and sorting operations.

Design Script also introduces the use of functional and imperative languages nested inside each other. This solves some of the restrictions of dependency loops that exist when you require to change a up-stream value based on down stream data. When used in a purely functional mode it allows for code to be mapped directly into symbolic or graph form or graphically generated code to be converted into Design Script code. In this way, it represents a programming language that encodes the directed dependency of traditional parametric models but in script form. By capturing the associations between variables the compiler is able to re-evaluate variables if those they directly depended on change. With the intent that users can use less code to describe complex concepts, with the aim to improve readability. Furthermore they can mix method of model creation between graphical programming and coding to suit the task or users' experience.

These innovations bridge the gap between graphical programming type interfaces and script based programming. This work has now been integrated into 'Dynamo' which is a scripting and visual programming interface to Revit Autodesk's BIM software.

2.3.1 Conclusions

Computational design represents a powerful method to control and generate geometry. However for most non-programmers, wielding this power has proven challenging to apply effectively and meaningfully. Nonetheless since its conception there has been significant progress in making this approach intuitive and understandable to designers. As a result the number of users of such systems has risen rapidly.

The important feature of these systems is its ability of abstracting away the

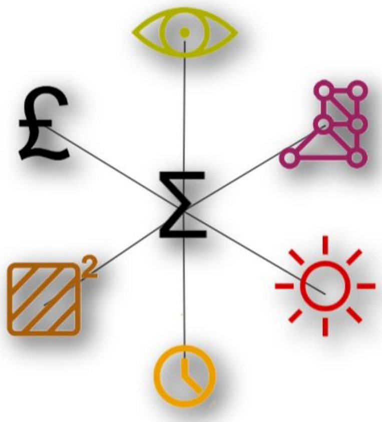


Figure 2-9: The quantitative assessment of design is the sum of its significant properties

design into a design generating logic. Essentially, the program is recoding the designs *intent* rather than any singular design. Removing the need to explicitly define geometry manually. This enables reconfiguration of such logic with relative ease of regenerating any output such as 3D model or drawings. Furthermore the computational capture of this logic allows it to be driven automatically, so versions can be generated quickly as well as used to optimise inputs as will be shown later.

2.4 Evaluation

A significant proportion of the design effort besides representation is in design evaluation. The critical assessment of the suitability of a proposal is paramount if designers are to be able to understand and improve on their design. Assessment typically comprises of analysis followed by comparison to desired criteria. Criteria can be varied; for example, explicit or implicit, quantifiable or qualitative. With different priorities based on the type of building, client requirements and who the end users are to name but a few. It is important to note here that there is more to evaluating a design than purely engineering analysis, for example, aesthetics of a design, the quality of view out of its windows, and so on.

The use of computation for engineering analysis has had a major impact not only on how analysis is undertaken but also on what is possible to design. Engineering analysis was one of the earliest commercial users of computers, being inherently numerical and gaining much from faster and more accurate calculations that computation enabled. As time progressed, hardware has improved bringing engineering analysis to desktop machines and common usage. This thesis focuses on structural engineering issues, as this is the area of practice of the researcher. However, much of the issues and conclusions are also transferable to other engineering fields associated with architecture.

2.4.1 Structural Analysis

The introduction of finite element analysis (FEA) has made solving complex interactions trivial. FEA methods rely on the solving of a global stiffness matrix with a size that is based on the number of nodes in the model, and comprised of the sum of the individual elements stiffness applied to the relevant connected nodes. Each element stiffness being defined by its shape function which relates to its geometry. These can relate to solid continuum like models, or frame models which represent larger structural objects like rods, beams and plates. Modern fast solvers integrated into finite element programs, have shifted effort from calculating results to creating the models and interpreting the huge range of results. It is still a significant challenge to interpolate between architectural and analytical models.

There is a fundamental difference between the representation of architectural and engineering models. Generally speaking architectural models being concerned with representing the visible surface whilst engineering representation is concerned with the relationship between elements and their physical-material specification. Some BIM software has presented ways to interconnect the separate systems, however, they are often restricted with vendors linking to their own software families but rarely supporting true interoperability. There is a ISO open standard for BIM

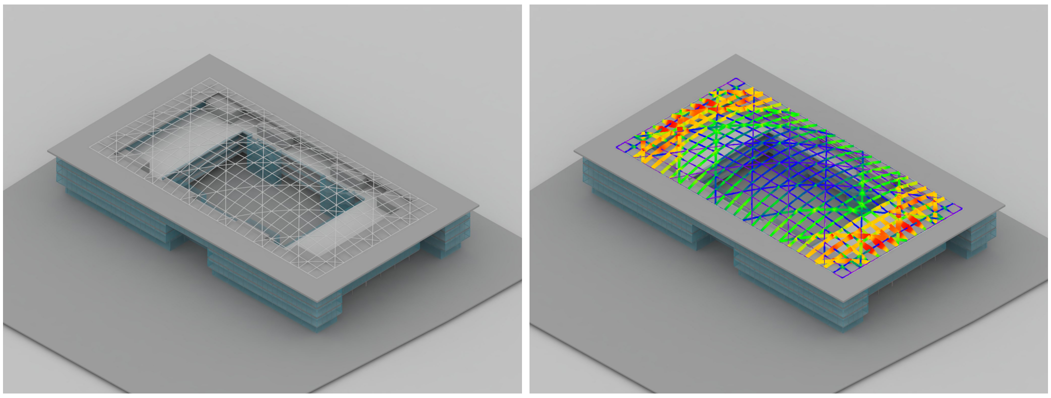


Figure 2-10: Cleveland Clinic Project roof option with superimposed Analysis. Source Foster + Partners.

interoperability: the 'Industry Foundation Classes', this has been in existence as early as 1994 initiated by Autodesk, however still this is not as well supported as Autodesk's own format for Revit. Both by Autodesk but also by other software companies involved in BIM as Revit has a very significant majority of the BIM user base. [4]

As such there are currently significant practical challenges in the transfer of data between engineering and CAD software. There are methods employed by engineers (including those in F+P) to overcome these problems, however, they are often very manual time intensive and crude in nature. An example work-flow being to generate simplified centreline models with element properties divided by layer and importing these into FEA packages. Whilst this is acceptable for rough calculations and basic orthogonal frames, this method becomes unacceptably imprecise for more complex geometries such as when beam section or orientation is unique for many elements.

[4] There is the potential for the Revit format to become its own standard. There would be a precedent in the wide use of Autodesk's .dxf format whcih is now widely used as a interoperability CAD format by many vendors. However the Revit format is significantly more closed being a propriety binary, thus this adoption would not be anywhere as open as the human readable .dxf format.

2.4.2 Integration

The typical division of labour and interest between the different design consultants involved in a project, understandably partially follows lines of legal liability. With structural engineers expected to solve and satisfy stability requirements for a design, environmental and M+E engineers concerned with the habitability of the space and sustainability, through to more specialised consultants such as acoustic engineering and people-flow analysts where required. With each owning responsibility in the design, architects as a result, can be quite remote from the performance and practical requirements of a building. The more professions and individuals included, the higher level of complexity, which can engender an environment with little overview and each group takes care of their own interests. There is significant benefits to an integrated approach for the final built design, and this is the driving reason behind the integrated design team in Foster + Partners. However the siloing of individual's roles and scope of influence is restrictive to an integrated approach, which is then mirrored in technology supporting people in those roles also.

2.5 Strategies to Integrate Representation and Analysis

Foster and Partners as a practice, takes a technological and performance-based approach to design. As such the strategic benefits of improving integration were identified early. This is why at an organisational level there was a decision that engineers should brought in to the practice. as a means to remove barriers specifically, legal, physical, social and working infrastructure. With these new opportunities, they also presented new challenges for novel solutions to improve and engender integration on projects whilst still keeping the quality that Foster + Partners

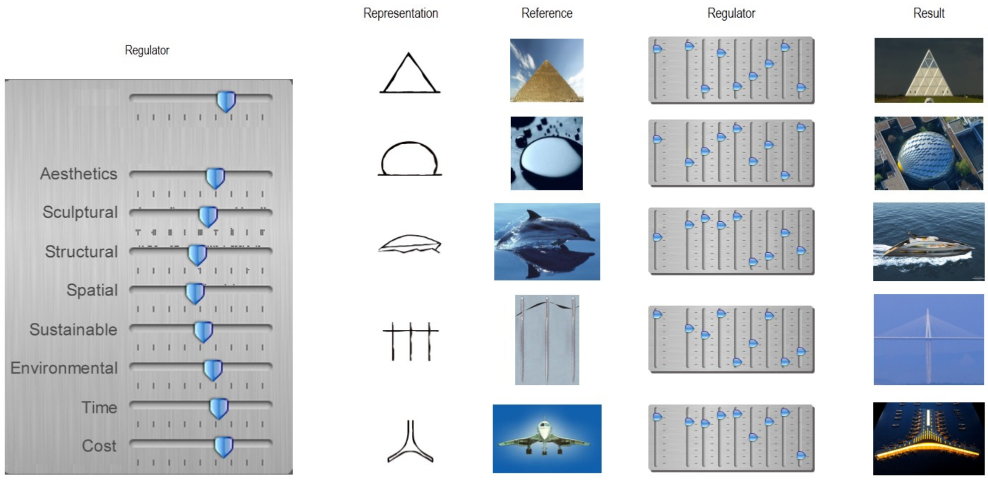


Figure 2-11: A proposal by Whitehead on determining designs by compositions of outcome sliders (eg Aesthetics, Sustainable, Time, Cost) rather than typical numerical geometric inputs [Whitehead and Josefsson, 2011]

are respected for. This was one of the key problems posed by the researchers appointment at Foster + Partners. To overcome the problem it involved understanding what exiting systems were in-place, understanding the issues these caused, sequentially proposing and implementing these solutions.

It is worth noting that whilst these challenges are in some ways unique to Foster + Partners due to its specific work culture, the majority of the issues addresses are essentially common between any group of engineers and architects working together. Furthermore the 'integrated practice' is not new with firms such as Arup, SOM and Atkins having engineering and architecture side by side in the same company. As such there is wider value in looking at how to support integrated working practices in the building design industry.

2.5.1 Integration at Foster + Partners

Foster + Partners work on a wide range of projects, with no two projects posing the same challenge. The requirements and constraints of a project imposed by client, site, context and program to name but a few, always generate unique problems.

One interesting formulation of design requirements within Foster + Partners is shown by Whitehead in figure 2-11, where it is argued that each project has a different order of relative priorities of the aspirations or objectives. Whitehead defines these aspirations as aesthetics, sculptural, structural, spatial, sustainable, environmental, time and cost. These aspirations are perhaps indicative of the clients and projects that Foster + Partners get involved in, which are often iconic and high performance. However this range of requirements would be common between practices, even if there are different objectives.

Teams are orientated around projects, and are comprised of a mix of architects and engineers that reflects the project's aspirations. For example, significant infrastructural projects would comprise of more engineering expertise, whereas conventional but high end residential would expect to have less engineers and more architects and interior designers. It is worth noting that this is typically only for integrated projects where the client has agreed to engage us for a complete 'turnkey' service. This allows engineers to co-locate and work collaboratively with architects.

Whilst this was initially quite rare in the practice these projects are now becoming more common, especially for competition and early stages where this way of working has the most benefit. It is on these integrated collaborative projects that the author was mostly engaged with, however in some instances work was carried out solely by the researcher but predominately in cases where the input required was more purely geometric in nature or the level of structural input could be covered by the researcher alone.

2.6 Structural Representation and Evaluation at Foster + Partners

Due to the close contact and high level of interaction of engineers and designers, many projects at F+P have encountered the problems of interoperability between engineering and architectural models. These issues often stem from problems of translating information between specialist programs. Thus it was identified early on that this would be of worth investigating. After discussing this with a number of engineers as well as some architects, there were some recurring themes drought up:

1. The conversion between architectural representation and structural representation takes a long time due to the manual processes .
2. The two models (CAD and FEA) are often developed independently due to the slow iteration time in engineering compared with architecture.
3. The engineering input can be out of date by the time it is ready, this is especially true for early stage design.
4. Architecture and engineering teams use more than one CAD/Parametric platform or FEA package respectively. Depending on what is preferred or best suited for the task, and these preferences can change over the duration of a project, thus conversion must be relearned, for each project or even between project stages.

The resultant survey findings mirrors similar experiences by the author. Thus, it was identified there was a need for a method to improve this situation. The primary aim was to have an automated way of converting between different software especially towards building structural models from CAD, which was identified as the most labour intensive conversions (not to mention mind numbing) under-

Figure 2-12: Example showing the number of links required to link programs if implemented as a one-to-one set of links or to a central hub. Source Author

taken. A review was taken of all the software platforms used for structural design integration. This was then filtered down to the most important links required which numbered seven. Two CAD platforms Microstation and Rhino, one parametric system Grasshopper, two structural platforms ETABS and GSA (although SAP2000 was also considered) and two data and programming interfaces Excel and VBA scripting. The level of individual conversions required for such a system was very high.

2.6.1 F+P Hub

Thus the idea of a conversion hub was proposed by the author as a solution to this issue. The conversion hub would act as a central representation of a structural model, which was linked to a series of classes programmed in C# as a Microsoft 'Dynamic Link Library' (DLL).

Each new translation (irrespective of being structural or architectural) could be written an extension to an abstract conversion class, thus connecting to conversions to any other software that already had a link written. It is worth noting that each link has a separate functions for import and export as these are quite different tasks depending on what system one is using.

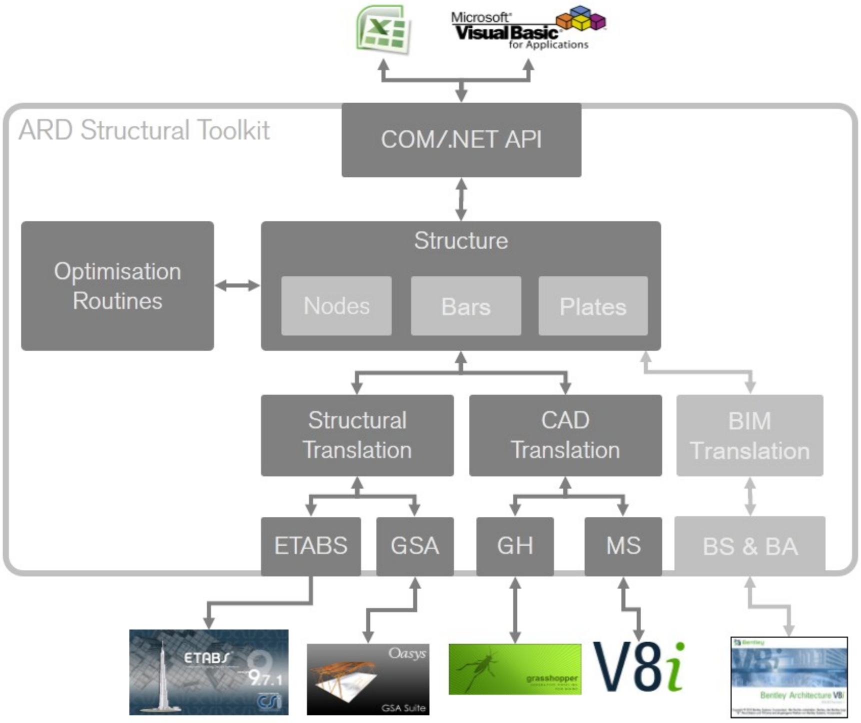


Figure 2-13: System diagram of hub connection with links to external programs and programming interfaces. Source Author.

Adding a link was similar to adding a spoke to the hub, rather than the alternate, which is creating custom conversions between each and every platform. This reduces effort from a potential $(n-1) \times n$ for single links to n for the hub, where n is the number of programs to connect to. The hub was written in C# as a platform agnostic DLL using the .net framework. This was chosen as it is the core of the Microsoft framework, which by being a part of the common language runtime enables different user-centric interfaces that utilise the hub to be easily written .

Microstation GSA Link

The first application of this link was a tool created to enable the conversion of data between Microstation and GSA (Oasys' structural analysis software), using the hub. This tool was written in native Microstation VBA, allowing it to be used by any Microstation user as a custom plug-in. In this case, geometric lines and points in Microstation were used to represent beams and nodes respectively. The design level (also know as layer) and colour mark-up of the geometric objects was used to denote beam sections and node fixity or loads in Microstation. This conversion was determined by a conversion table that related the colours to specific sections etc. This table could be loaded or created by using the tool for that session and saved allowing users to reuse conversion conventions easily.

This complemented the previous methods, to bring in CAD geometry to GSA, but rather than relying on the manual process it was able to instantly build a GSA model. This was especially useful in cases where various tries were required to get the conversion correct, and proved to be a faster interface than that in GSA when changing complex combinations of beams in a model.

This in memory representation, rather than a static file based conversion, has further advantages as it is able to generate a persistent link to the structural software, which opened up the hub, to be extended to then extract analysis data once the model was solved. In this case this was used to round-trip structural results data from GSA back into Microstation. Enabling engineers to capture their data in the same environment as architects. This meant that the results could be introduced in renders and placed alongside architectural visualisations was more easily achievable. This was found to be a significant assistance to presenting the findings and conveying the importance of such studies whilst in design meetings. Countering the unfortunate but understandable trend in design meetings of people to engage and discuss topics which are better presented more than those which are less visually engaging. This trend is the opinion of the author but also shared by others, engineers and architect alike, and can leave important engineering issues under explored.

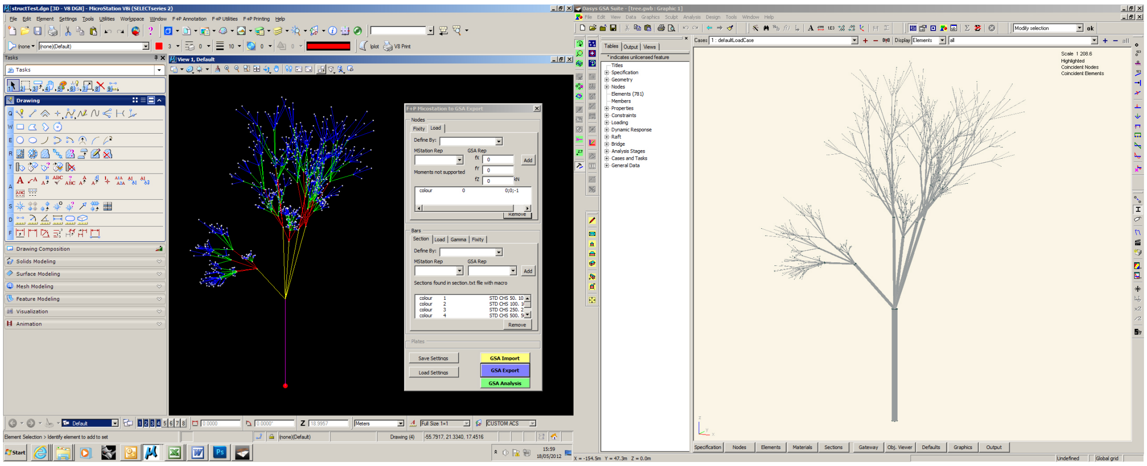


Figure 2-14: Example use case of exporting centreline geometry from Microstation to GSA via colour mark-up. Source Author.

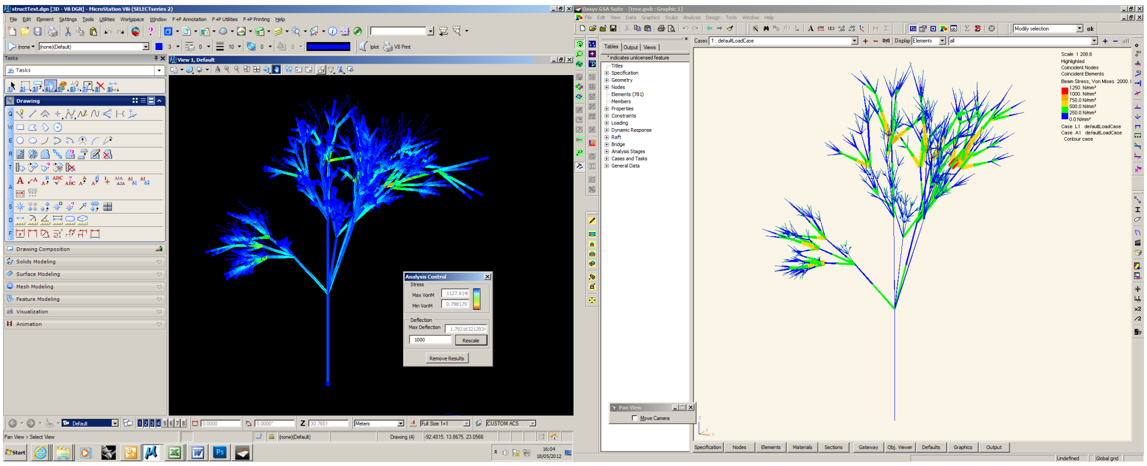


Figure 2-15: Example production of visualisation of GSA analysis showing von Mises stress and deflections in Microstation CAD on left from the GSA analysis on the right. Source Author.

Grasshopper Link

The same hub was integrated to produce a Grasshopper plug-in specifically for use by the ARD and E01 groups. This was written with the aim to create structural models and also bring engineering data back into Grasshopper. A plug-in was developed with a series of components which allow for the creation of structural elements. The system is based on developing full structural frame models from centrelines.

This included the ability to create beams by input line, input normal vector to define beam axis, and a cross section. Structural objects which are not represented geometrically, such as fixity conditions and loading also had custom components. These could then be linked to beams and nodes to augment their behaviour. The model elements, as well as the global properties such as load cases could then be passed into an analysis component which in turn could generate a GSA model and optionally return the results back into Grasshopper.

Being a plug-in to a parametric system it is possible to build parametric models which generate analysable structural models. These can then be changed by modifying input geometry or values and have the model automatically update the model and any results.

This plug-in was extended as functionality requirements emerged. As a result, over 24 components have been created covering a wide range of model creation. This also drove development of the hub, when it was not able to provide the necessary method.

2.6.2 Integration in Practice

After the creation of the F+P structural tools, they were applied to numerous projects. In the majority of cases, this involved quick structural model creation during con-

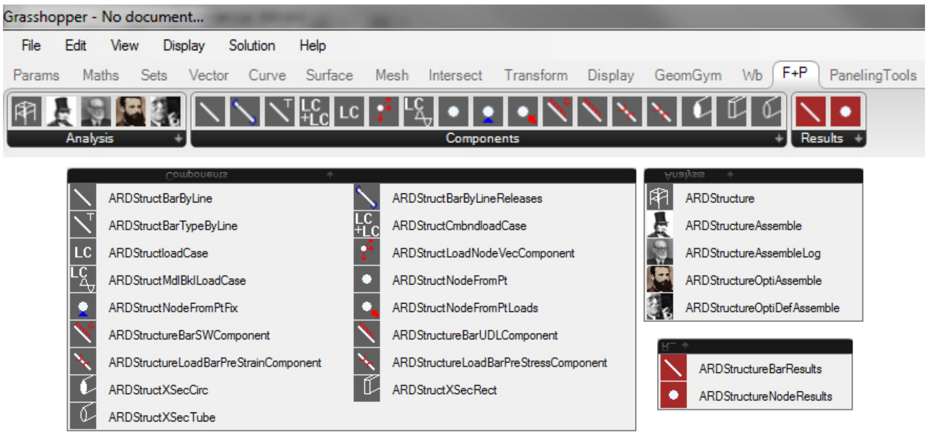


Figure 2-16: Components that make up the F+P structural tools in Grasshopper. Source Author.

cept or scheme design phases. The tools have not been used in protracted long term project use, but there have been a number of intensive extended sessions of use.

The application of the Microstation tool was limited to some trial cases but was used directly by those in the engineering group. The Grasshopper link was applied much more especially by the author and other team members. Due to a lack of parametric design experience by the engineering team, they were either more likely to have the ARD group drive the model or manipulate the sliders, but not set up or modify models themselves.

The tool has been applied almost exclusively for early stage design phases. This is most likely due to the fact that this is the stage where the author and the author's team have the most involvement in projects at the practice. However it could also be because this is where the tools are were most effective, and the project requirements match the current level of provided functionality.

Where it has been used the principal applications that it was used for are:

- Understanding how the structure acted.
- Quick sense checks for the viability of a design option to work with reason-

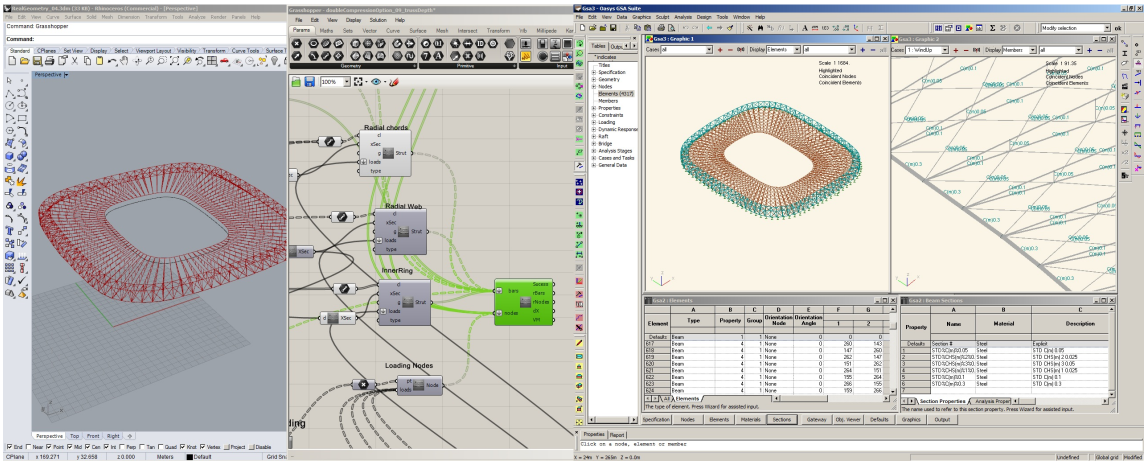


Figure 2-17: Example of work session on bicycle wheel roof model for Madrid FC stadium. Source Author.

able/desired sections.

- Generating stress plots to explain the behaviour to Architects.
- Comparisons between options, to differentiate their structural efficiency.
- Calculations to see if boundary forces where reasonable for interacting with other parts of the structure.

One in-depth case study is the Madrid FC stadium redevelopment proposal. The scheme involved placing a new roof structure on-top of the existing stadium bowl as well as expanding the west stands to include better hospitality and non match day entertainment and facilities; This was to bring it in line with the new FIA standards.

The new regulations called for a significantly larger roof coverage over the pitch but rested on the same supports for the original smaller roof. The design progressed through three main options: an arched truss, a bicycle wheel type cable structure and a cable net based hyperbolic paraboloid. In all cases designs were generated via a parametric model in Grasshopper by the author and generated by the same system in GSA. It was also possible for this to be seen as CAD geometry and capturing key views with the rest of the scheme, allowing for visual appraisal.

During these design-sessions, the model's parameters were often manually 'tuned' to find something that seemed efficient by changing the values and observing the results in real time. In some sessions, it was possible to dynamically change values to understand the relationship between performance criteria and the changed value, creating an intuitive understanding of the sensitivity of the design space.

During the process, it was found that support was required for pre-strain loading for cables and multiple load cases as the governing load case was ambiguous, but due to the systems architecture, this did not take long to implement. This system was able to offer a much faster turnaround time between modelling and analysis of each option, typically needing less than a minute to generate a new model and results, rather than the twenty minutes plus that was required for a manual conversion let alone the geometry creation, leading to a more refined result compared to manual methods.

In some cases this enabled focused design sessions where engineers and architects proposed new values of parameters for an option, which were then generated in real-time, commented on both structurally and visually before being tuned based on that input. These rapid feedback loops were in part due to having a representation where both parties are able to get the data they wanted in a platform familiar to them, thus fast-tracking decision making.

2.6.3 Discussion

The development of the F+P structural tools and conversion hub highlighted the improvements that could be had by integrating representation and analysis together dynamically; Both in the performance of the resultant design and any considered options, by enabling users to set up and test options and assumptions quickly, resulting in more design iterations, but also in the process of design by promoting the use of parametric systems to model these geometries. It prevented time consuming loops of manual modelling for analysis, which although it can be as little as thirty minutes to the progress, it can also be as much as adding a day. However more importantly it removed systematic breaks or pauses in the immediate design exploration process leading to the design loosing momentum.

There are some issues encountered by the author whilst developing the tools which enable this approach. This method is reliant on any linked program having an exposed application-programming-interface (API). This was not the case in ETABS at the time of development. Whilst it is typical to have an API for most software, (this has since been resolved for ETABS), it restricted the usefulness of the tool in this case. The hub also operates with a live copy of a program and holds everything in memory. As such a valid licensed copy of any program to be linked is required to work. This could be an issue if architects wanted to use this software as they are unlikely to have the structural software. This is not the case for methods that translate by exporting files for each program, however this does not allow for dynamic round tripping of data. Although the hub represented a singular structural representation with which all external programs can then link to, to reduce code. Owing to the different properties of each interfaced program, such as layer and colour for CAD or element number or group for FEA, this still required significant expansion of the hub code to support this. As a result, the hub code represents over 4000 lines of code, with the Grasshopper interface representing a further 3800 lines.

2.7 Alternatives for Integration

Other practitioners have also identified the potential of integrated parametric and structural systems. One very similar tool is for Grasshopper called 'Geometry Gym' by Mirtschin [Mirtschin, 2011]. This is a suite of tools that primarily gen-

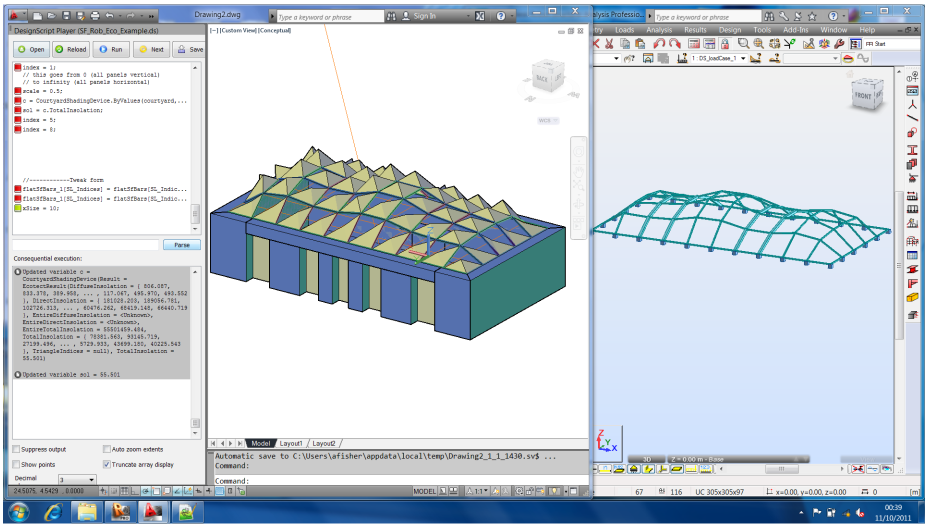


Figure 2-18: Example of early Design Script environment linking script to Autocad geometry and structural model in Robot FE. Source [Aish et al., 2012].

erates IFC BIM files, but also GSA models directly using GSA. This was initially not considered as it was oriented solely at Rhino3D and Grasshopper and ignored Microstation an important platform for Foster + Parters. However, more recently Geometry Gym has begun to have significant usage, in most cases in place of the hub developed by the author. This has been encouraged by the author as it has documentation as well as professional support helping users get to grips with the software with greater ease. Geometry Gym adoptions also has helped in reducing development load from the author, with feature requests by a project team often coinciding unfavourably with the authors support of that same project.

Another alternative that has emerged is Dynamo which is a parametric system for Revit, producing BIM models that can be linked to structural software, by plug-ins to link to specific analysis software. However this option was prohibitively convoluted as Revit was not widely used at the time. The author was previously involved in the development of Design Script which is now being used as a basis to Dynamo, where a major component of the work was devising a link between the system and Robot a FEA program owned by Autodesk. However, it is worth highlighting that here again the issue of a platform specific tie-in reoccurs. As links to other analysis platforms which are much more widely used are currently not present.

2.8 Conclusions

This chapter has shown how computation has firstly mimicked design representation by copying the drawing board methods, but now how more computational methods and understanding of generating design representation is becoming widespread, specifically with parametric modelling. These methods have empowered a higher level of experimentation and flexibility with design parameters.

Equally, integration methods have progressed on top of this technology to complement the new flexibility with data about the options considered. These are able to change how we link engineering and design representation, by speeding up model creation and linking to a model that can also be easily changed, many more options can be explored with the engineering results. This has shown to improve collaboration and the quality of design.

Chapter 3

Rationalisation and Optimisation

> "The most advanced chapters of theory of structures ... can only be used to check the stability of a structure. They can be used only to analyze numerically a structure already designed, not only in its general outline, but in all its dimensional relations. The formative stage of a design, during which its main characteristics are defined and its qualities and faults are determined once and for all (just as the characteristics of an organism are clearly defined in the embryo), cannot make use of structural theory and must resort to intuition and schematic simplifications."

Pier Luigi Nervi, Strucutres, 1956

Parametric and computational design has enabled a new level of sophistication for representation and analysis. This in turn has enabled greater freedoms and possibilities in design. However equally it poses new challenges to rationalise and improve these designs. Whilst designing as an activity is concerned with the definition representation and evaluation of options, it is also concerned with improving their performance over the period of design development.

Historically the number of options capable of being produced and considered was in direct proportion to the amount of human resource creating them. Complexity or sophistication is also an issue with complex designs potentially able to perform better than a simple one. This notation of complexity being required to describe or solve complex problems is explored by Ross Ashby with his 'Law or requisite variety' [Ashby et al., 1956]. Typically, sophisticated designs take longer to develop and so cannot be iterated as many times, acting as a barrier to making complex but well explored design options as mentioned by Rittel and Webber [Rittel and Webber, 1973].

This trend has been reversed with the introduction of computational methods, which have the potential to make design process which at best decouple complexity and generation time. This is especially pertinent in the case of option exploration with the use of parametric design, which was expressly developed to support this way of working. These methods open up possibilities to improve the design more than the basic manual evaluation cycle could allow, again by employing computation to actively drive parametric models, with algorithms to improve the design by acting on the geometry via the parametric model. This chapter will show some common themes in this area of research, and the innovations and findings of the researchers efforts applying this to practice.

3.1 Rationalisation

It is desirable and often required by both client and other stakeholders for a building to be economical and efficient both in resources and construction cost. In general there are two approaches to this:

- **Post-rationalisation** the process of changing a design from an initial flawed starting point to improve it, often with as little visual change as possible.

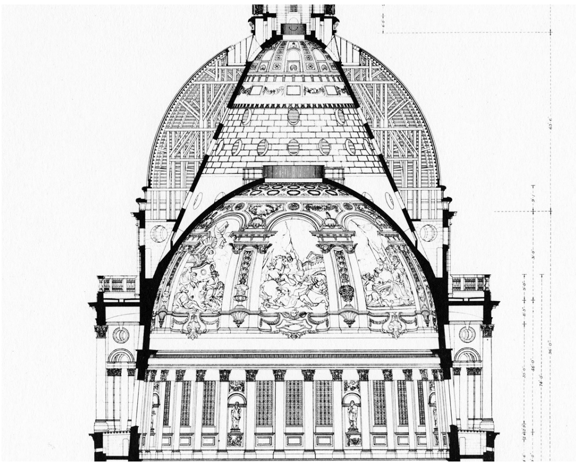


Figure 3-1: Section through St Paul's dome London showing rationalisation of the form into three separate functional requirements; a aesthetic exterior lead roof supported by a masonry structural shell held at the perimeter by a chain, with a free standing interior dome. Source Wikipedia.

- **Pre-rationalisation** where an initial design process is constrained in a way that it produces well performing designs.

Often the decision to undertake either of these approaches is not something that is chosen but stems from the requirements of the design and the interests of the collective design team, as well as the emerging requirements. Post-rationalisation is often needed after initial design work to realise something that is desired but impractical and or expensive. For this reason it is mostly applied later in the development process. The improvement can often be measured quantitatively rather than the often more qualitative decisions and judgements earlier on in the design process. The metrics are often externally defined as they are fed back from consultants such as engineers for compliance, or from fabricators and the metrics are needed if the design is going to be in budget or even feasible. Frequently geometric factors are translated to other more project-centric metrics such as cost, time or quality. Economic factors are often not as central as one might expect, as by the time rationalisation is discussed, contractors have typically been appointed, often already with a fixed price contract. By this stage the emphasis is on working with them to get the best result completed within the deadline.

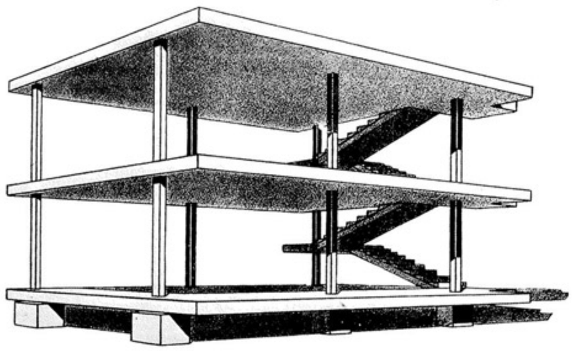


Figure 3-2: Example of the pre-rationalised 'Domino House' efficient structural module by Le Corbusier, that was integrated into many of his designs. Source Wikipedia.

Pre-rationalisation on the other hand often comes from the desire to achieve certain design goals from an early stage. It often involves a strategy or a series of restrictions, which if implemented in the design process are known to make it perform better. These may realise objectives in terms of material usage and/or physical behaviours such as compression only masonry domes. It may have aesthetic or geometric requirements such as to have no orthogonal elements. The emphasis in all pre-rationalisation cases is that the design goals are identified early on, even before design has commenced, with the design process modified to realise these requirements. Methods applied to realise these goals vary, but can be as simple as using predefined modules. However recently it is often easier to encapsulate these processes computationally and drive them with parametric modelling, as this does not adversely encumber the design process and can provide feedback during the design development.

3.1.1 Post-Rationalisation

From an implementation point of view, post-rationalisation is concerned with converting a pre-specified design into one that performs better but still maintains the fidelity of the original. By definition post-rationalisation occurs after the main design decisions have been determined. As such, many of the concerns are aligned with practical goals of a project to be constructed. Often issues are quite specific to one problem but not always. Areas of construction issues and thus frequent post-rationalisation efforts are; flat-panel-glazing, structural frame simplification, reducing geometric torsion in beams, increasing a design's repetition and reducing the number of connections in glazing/structure. Post-rationalisation often is concerned with describing one 'target' geometry with another of a lower complexity, typically due to manufacturing and construction constraints and any materials used that aren't orthogonal and/or planar attract more cost. These methods can be divided into one of two approaches:

1. Firstly, by putting effort into interfacing with better methods of construction and giving the required extra data to enable its use.

2. Secondly, by simplifying the design's construction by using more conventional means but in a way that does not adversely affect the final outcome.

Improving the interfacing and sophistication of construction methods in order for construction industry to is an area of significant current academic research. Examples include devising structures that can be laser cut out of 2D elements and packed efficiently [Dritsas et al., 2013], the application of robots for off and on site fabrication [Gramazio and Kohler, 2008], the robotic creation of blocks for masonry structures [Bärtschi et al., 2010], even the proposal and demonstration of aerial drones for construction [Willmann et al., 2012]. There are a growing number of commercial companies helping to span from architecture to construction with industrial robotics such as Design-to-Production, Gramazio and Kohler and R-O-B Technologies. The author has been involved in consulting on arguably one of the most extreme proposed applications of such technology, the application of robotic construction on the moon using 3D printing of stone [Ceccanti et al., 2010], whilst at commercial engineers Buro Happold.

However, besides from laser cutting, robots are not widely used on typical buildings especially those of a reasonable scale. As such, the much more typically applied solution at present is the simplification of the construction geometry. There equally have been some major advancements in this field, especially with geometric treatment of glazing issues leading to sophisticated mathematical methods to derive planar quad meshes [Liu et al., 2006] [Zadravec et al., 2010] from an arbitrary base surface. And extensions where to match base forms to similar developable surfaces which are easier to panellise [Flöry et al., 2013].

This is especially relevant in practice for some architects who specialise in 'free-form' architecture. For example Frank Gehry whose deigns are often conceived initially as card models, and then expected to be faithfully reproduced at building scale by the rest of this practice. This requires a considerable amount of post-rationalisation and there is a large group dedicated to research and application of this within that practice [Shelden, 2002].

Similar processes are undertaken for the geometrically complex work of Zaha Hadid's studio, often by external engineers [Kaijima and Michalatos, 2008] or specialists [Pottmann et al., 2008] and recently internally [Bhooshan et al., 2014]. Here however, the forms stem from an interest in forms uniquely defined by digital tools and these are actively used to challenge existing design approaches. There are also external consultancies such as Evolute and CASE Inc whose commercial offering centres on assisting architects to rationalise designs to be constructable, either by facilitating with advanced mathematical rationalisation of panels [Eigensatz et al., 2010b] or by simplifying processes or definitions.

Post-Rationalisation at Foster + Partners

At Foster + Partners there is often the need to rationalise forms derived from functional or aesthetic requirements at a bigger scale, but then interpret them again to resolve construction issues. An example of post-rationalisation undertaken by

Figure 3-3: Visualisation of UAE Pavilion for the 2015 Expo Milan; showing overall scheme and view of the sand dune main 'canyon'. Source Foster + Partners.

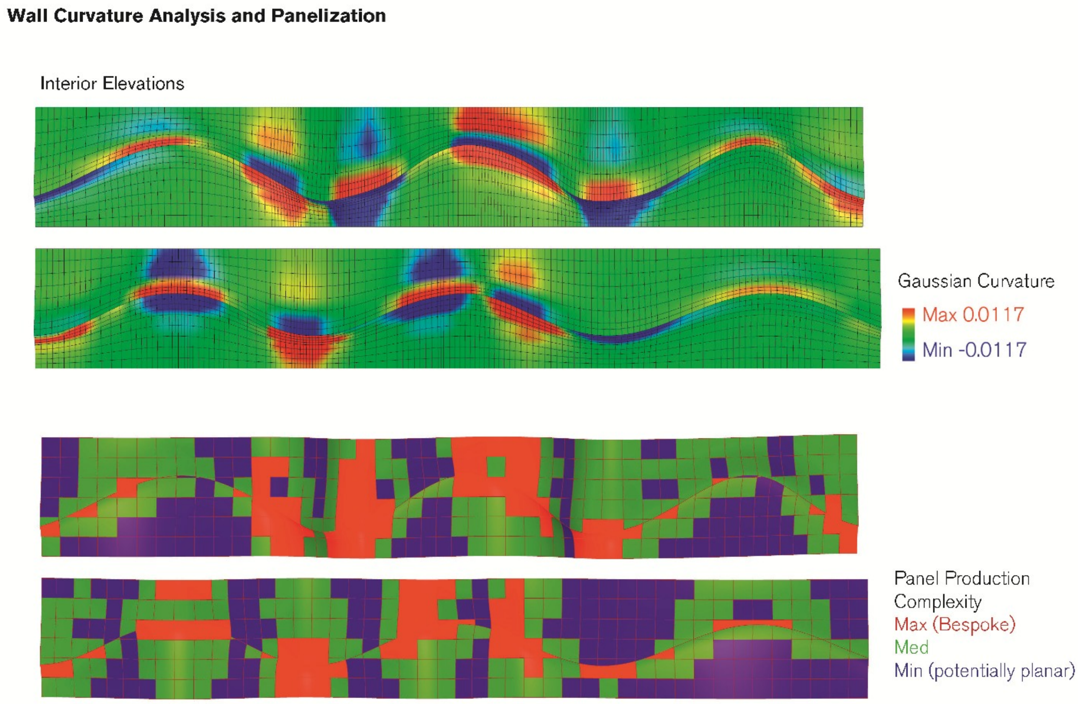


Figure 3-4: Initial panel analysis for the 'canyon' section, elevation is unrolled.

the author was the UAE Pavilion for the 2015 Expo in Milan. For this project, the concept was to evoke the complex sand dunes of the Emirates on the walls of the pavilion. This represented over 7500m^2 of proposed wall area, and a major visual component of the building. After making numerous design options, it was decided that the sand dunes properties would be reproduced at three separate scales: In plan, by the undulation of the walls, in the section of the canyon walls and on the texture of the surface. A visualisation of of the concept can be seen in figure 3-3.

The primary role the ARD team was tasked with was to produce the geometry for the UAE project. First steps were to generate the canyon geometry. This followed the plan curvature with basic vertical extrusions on external façade and with a large ridge line on the internal 'canyon' section.

To realise this complex geometry, glass-reinforced-concrete (GRC) was pro-

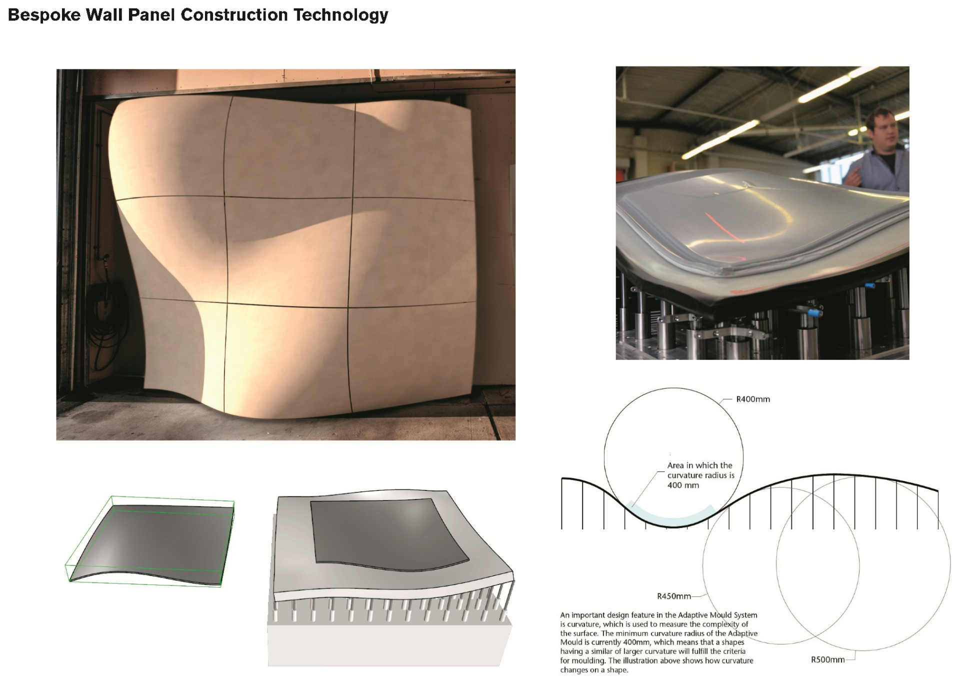


Figure 3-5: The Adapa mould system with actuated bed (top left), example panels (top right), and curvature considerations (bottom). Source Adapa.

posed, to capture the material and the desired mould complexity. At a very early stage of the project at the start of 2014 the buildability of such complex geometry for a project with a short lead time panel geometry was considered critical, especially with a completion date for the whole design in mid 2015. An initial analysis of the panels was undertaken and it was found many would need to be double curved, which was well known to be expensive for traditional mould making [Eigensatz et al., 2010a], and this was echoed upon interviews with potential contractors.

To overcome these problems it was proposed to employ a new mould making technology developed by the company Adapa, which employed special mould machines with actuators to produce adaptive form-work for the GRC. This could enable many different panels to be produced at a low cost and with no increase in price for complex curvature. Thus, the design was progressed on the basis of fully unique panels.

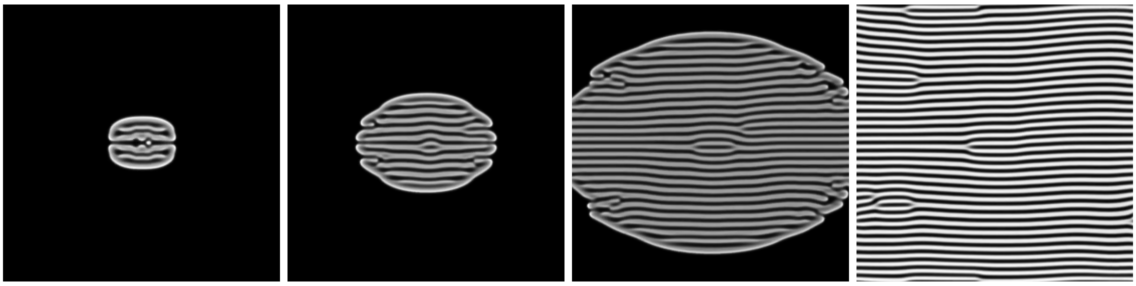


Figure 3-6: Example evolution of the reaction-diffusion method used to model the sand ripples.

However, owing to the emergence of uncertainties in the application of the ripple texture, paired with a very tight delivery schedule, it was later decided that a more traditional method would be preferred. To keep cost down, a strategy to use sets of periodic ripple patterns was proposed and implemented by the author. Initially a Cellular Automata method adapted from the work of Alistair Turner was used to mimic sand deposition, this proved to be too big in scale and instead a reaction-diffusion method was used, as it could produce the bifurcating ripples desired. Based on the success of this during design reviews this method was then significantly extended by a fellow member of the ARD team, resulting in a bespoke un-isotropic reaction-diffusion system, which used fixed periodic boundaries based on an initial free but periodic panel. This geometry was then reduced to an edge representation and given an analytical sand ripple profile in the valleys between the ripple edges or crests. This provided a family of non identical patterns which matched on the boundary of the panels. For the physical moulds this detail would be produced by a flexible mat placed at the bottom of moulds to change the face profile.

The initial wall geometry in plan that was based on a b-spline, was re-represented as a series of arcs and straight lines, in a process that will be described in detail later in the thesis. However this resulted in the number of arcs required and the allowable off set of the new plan from the old where minimised, creating a more rational build-able set of panels.

Figure 3-7: UAE Pavilion inspiration sand dune image with surface ripples. Source Foster + Partners.

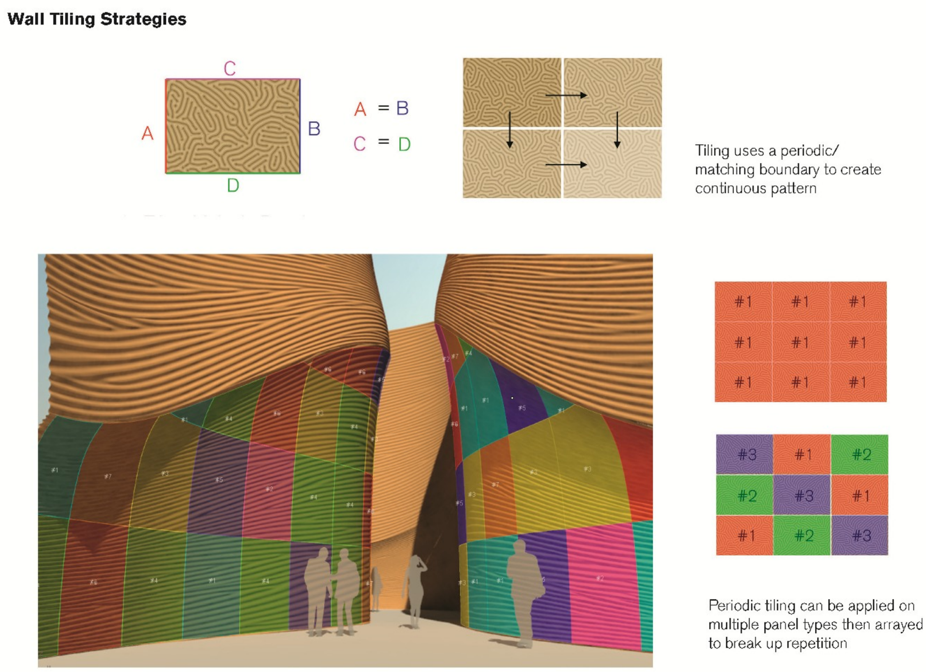


Figure 3-8: Example of early wall tiling strategy study. With each colour representing a different panel ripple pattern applied to unique geometry to make up the whole of wall. Source Foster + Partners.

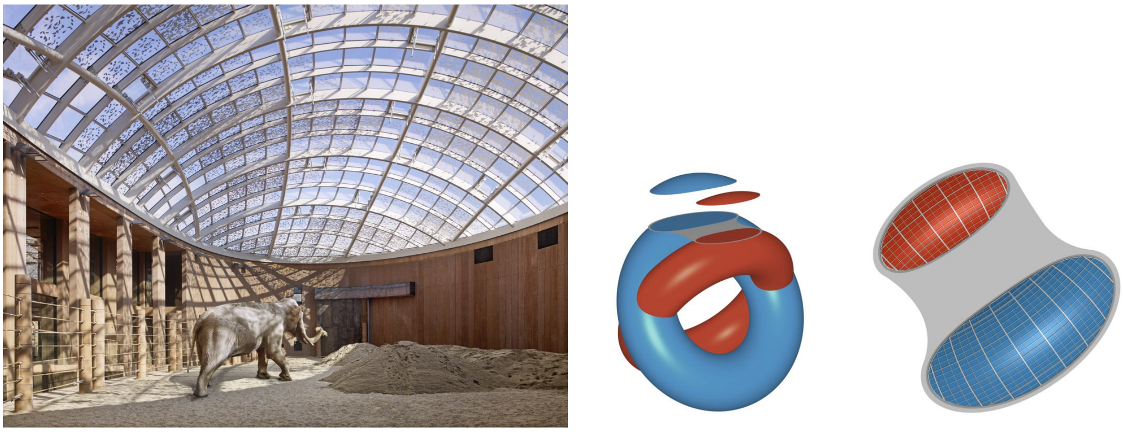


Figure 3-9: The Copenhagen Elephant House, an example of typical pre-rationalised design by Foster + Partners. Using two torus patches as generating geometry, a roof with efficient planar quadrilateral panellization with constant curvature beams can be generated. From [Peters, 2008].

Whilst post-rationalisation is undertaken at Foster and Partners it is actively avoided if possible. These methods are deemed only worthwhile when trying to deliver a very specific look that can not be achieved economically any other way. There is typically resistance in the office against forms that are difficult to construct if there is no underlying reason to have them, either functionally or culturally.

3.1.2 Pre-Rationalisation at Foster + Partners

Rationalisation is an approach that is often applied within the practice; derived from an agenda that it is desirable for a design to be intrinsically rational and practical for it to be successful. Typical examples are cases where performance aspirations are stated before the geometry is proposed. With the success of the design measured by the design goals not and not just the formal aesthetic result. From there, care is taken to ensure any geometry put forward to solve the problem/requirement is buildable.

The Copenhagen Elephant House is a good example. A pair of roofs were required to enclose the projects two main spaces and the resultant geometry was proposed a set of torus patches [Peters, 2008]. This geometry has often been utilised in projects, as it has a clear compact definition, is easy to set out on site and inherently has good curvature for structural loading via shell action. Torus patches, if gridded correctly, create planar rectangular panels and can be offset and still maintain planarity. The resultant beams between the panels, if rationalised into linear elements have zero twist or geometric torsion between the nodes. Or alternatively following the base geometry have continuous curvature making them simple arcs which are relatively trivial to fabricate.

It is of value to compare this approach with the PQ meshing methods devised by Pottman, where sophisticated computational rationalisation is required to obtain the same result but on any arbitrary geometry. However PQ meshing still typically generates panels of much varying sizes and often require nodes of different valence or umbilic points [Schiftner, 2007]. This is a much harder task and has advantages, but Foster + Partners by being in a position to drive the initial definition to something with inherently good performance such as torus patches at an early stage avoid the need for a post-rationalisation phase. In this example due to its fabrication simplicity it enables more contractors to bid and subsequently driving down cost.

The author has been involved in numerous projects where solutions have been proposed and generated based on well founded options. One such example was the potential use of a Geodesic Dome for a 'Sky Garden' on top of the Liberty Tower in Philadelphia. This solution was proposed by Norman Foster on the basis that this geometry would be effective in containing space efficiently whilst resting the wind forces present at that height. This proposed option was then translated into a parametric model by the author which could be fitted to the top tower. The model explored the scaling of the dome in different directions and compare with internal program sizes such as floor-plates. Different griddling was also investigated both triangular and the mesh-dual a hexagonal griddling, with the knowledge that this could be construed effectively because of the examples of prior art

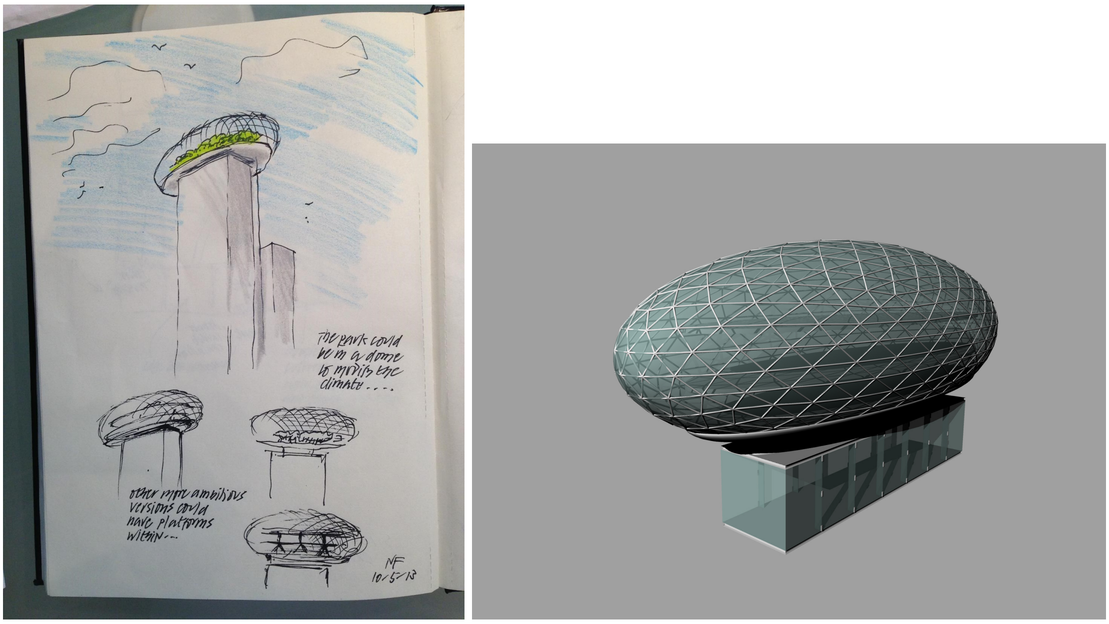


Figure 3-10: Initial sketch of proposed tower 'sky-garden' option alongside one iteration of the parametric model of the geodesic dome. Source Foster + Partners.

in geodesic domes [Teixidor, 2007]. Whilst this design option was not progressed in favour of another option more fitting to the overall building's aesthetic, it is an example of effective pre-rationalisation, by adapting an already understood and high performing system to a design.

Whilst these methods are effective, these 'off the shelf" solutions are not always appropriate, especially, when one is trying to integrate multiple concerns or face with a site or brief where a typical or conventional solution is not effective or too simplistic for the requirements.

3.1.3 Structural Rationalisation

One way of ensuring design has certain performance properties whilst still being relatively flexible is by creating approaches which only generate options with the desired performance characteristics. Some of these methods we will explore are those which are specifically concerned with structural performance. For design

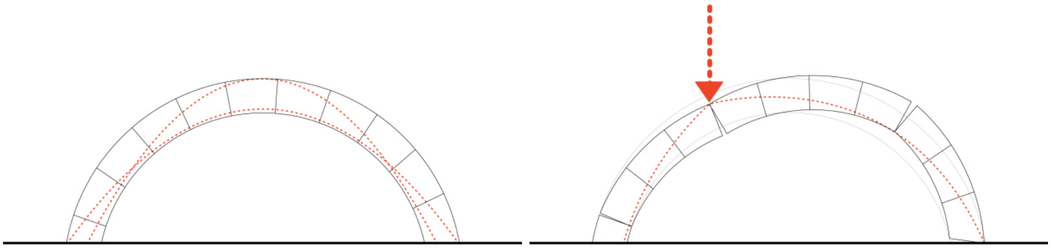


Figure 3-11: Example of thrust lines acting inside and outside of an arch leading to stability and instability respectively.

systems, there is a challenge to implement these approaches for a design context, so they are able to both provide a definition which returns the right geometry, but also flexible to design with creatively; freeing designers to move away from static singular solutions to ones which integrate more easily into the rest of the scheme.

These methods are differentiated from conventional engineering design; where conventional engineering attempts to solve problems already designed, structural rationalisation methods takes a specific approach to rationalisation which both directs and confines the potential options. This is in the pursuit of efficiency but also has the effect of changing the way design is undertaken. With these methods generating an efficient result often in a timely fashion, it becomes the role of the designer (whether engineer, architect or both) to manipulate their input to obtain the desired outcome, with respect to the requirements not rationalised for.

Dynamic Relaxation

An example of such a method is in the design of compression only shell-structures which are most effective when the shell depth is thin relative to span, relying on double curvature to transfer forces in the plane of the surface as opposed to bending as is the case with trusses or beams. Whilst any double curvature will promote in-plane shell action, the transfer of loads to ground is a non-trivial problem. Any thrust lines outside of the material will generate bending, which if significant in scale with deform the structure and in the case of buckling result in runaway failure and potentially collapse. Not solving this problem was a major contributing factor of the failures of numerous churches in the middle-ages.

One effective solution to the description of these forms stems from the observation of the Architect and Physicist Robert Hooke in 1671:

> "As hangs the flexible chain, so inverted stand the touching pieces of an arch."

This is because a chain is not able to resist bending forces, instead changing its geometry until the forces are resolved geometrically. Similarly but in reverse an arch following the inverted chain profile will exhibit no bending if it has the same uniform weight distribution as the chain. This observation can be used as a design method whereby the chain's hanging shape can be used for the definition of an efficient compression structure under self-weight. This physical phenomena can be exploited not just for chains but more complex networks of elements to aid the definition of many types of structure. Importantly this method allowed both efficient physical solutions alongside creative flexibility and control.

This was heavily exploited in the work of Antoni Gaudí where he used hanging chain models to develop efficient structures for his buildings. Most notably, to represent the overall design of La Sagrada Família. For this building he extended the concept using string with additional weights instead of chain to more precisely control arch lengths and accurately model the loads from the masonry and ornamentation. The string positions were then converted into the geometry for the principal compression arches and vault structure [Burry and Gaudí, 2007].

Later the work of Felix Candela, Luigi Nervi, and Eladio Dieste who developed mathematical descriptions of these surfaces finding special subsets which could be built or reinforced. Whilst this offers greater benefits it also constrains design especially early experimentation as these surfaces take a while to invent/uncover and

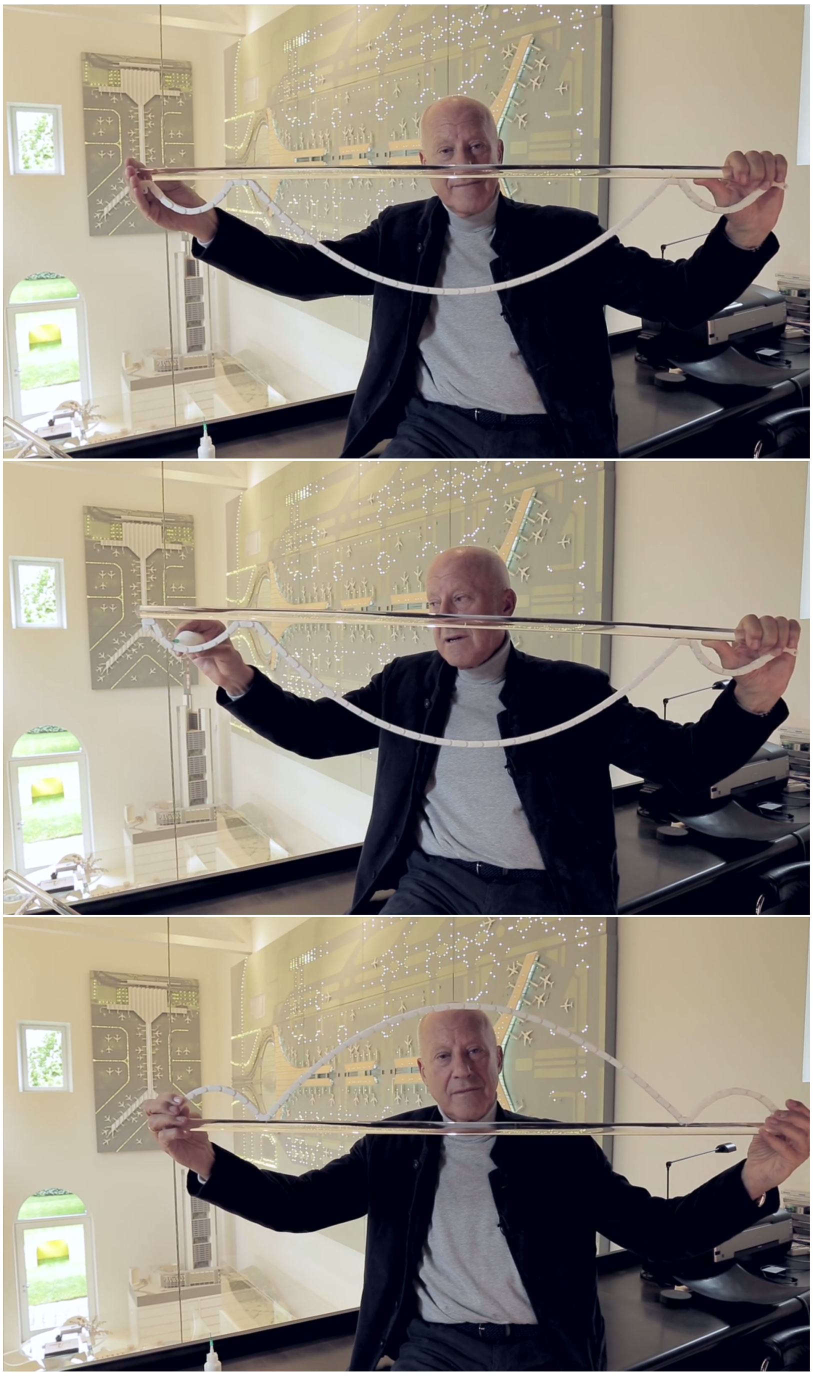


Figure 3-12: Norman Foster explaining the hanging chain principal. From the press and public video explaining the design principles of the Mexico City New Airport. Source Foster + Partners.

are not easily manipulated during the design process [Adriaenssens et al., 2014].

The collaboration of Frei Otto and Ted Happold as well as others during the 60's and 70's progressed the physical model as an intuitive design tool as well as analogue computer [Addis and Walker, 2005]. Using models to explore many different possible forms of compression and tension only structures, as well as tensegrity which combine pure compression and tension elements together. They pioneered the the use of physical models for determining forces by scaling materials for the correct stiffness and applying stain gauges to the model, carefully scaling the measurements to determine predicted full scale stresses [Otto, 1978].

At the same time and in tandem with physical modelling, the first computational approaches to imitating these systems where created. Using nodes with a notional mass and momentum, linked by springs and struts obeying Newton's and Hooke's laws, stimulations could be run, deriving the resultant geometry of the system for visualisation, structural analysis and fabrication. Much of this stems from Alistair Day's dynamic relaxation which using similar methods as varlet integration calculates the movement of a non-linear structure over time without needing to solve a large stiffness matrix unlike FEA methods.

On this basis, new computational tools to model hanging chains have arisen. These benefit from the fast processing of model hardware and 3D geometry developments in software. Examples include interactive modelling environment [Kilian and Ochsendorf, 2005] which represents a program written explicitly to work with dynamic relaxation which has menus to control features of the system such as gravity or individual spring stiffness. Other systems work within parametric systems, levering and extending their functionality; Kangaroo is a significant contribution to this field [Piker, 2013]. Kangaroo allows for real-time relaxation of geometry either seeing the steps or showing the final static outcome. Updating if elements are changed live by the user or fully rerunning when the number of elements is changed. The author has experience in both using these tools for design, but also contributing to the base theory and implementation approaches. One example is the generation of logic to create models of soap films which can respond to topology changes whilst still maintaining their minimal surface properties, by modelling the surfaces as a large number of freely moving but attracting particles representing the isotropic soap film stresses, rather than predefined defined links and nodes of usual spring systems [Williams et al., 2014].

Graphic Statics

Graphic Statics is a graphical force-analysis approach, attributed to Luigi Cremona [Cremona, 1890], as a method of solving truss or pin joined axial structures. It makes use of a force diagram, with each axial element represented by an edge with the same vector direction as the element, but a magnitude equal to the force in the element. Thus elements that have more force acting through them will have longer edges on the force diagram. By introducing forces/loads at nodes and points of fixity, which provide extra edges on the force diagram, 'force polygons' are created, which if closed, represent an equilibrium condition of external and internal forces at that node. In the case of statically determine structures this can be solved using linear optimisation or numerical methods similar to dynamic relaxation. The Thrust Network Analysis method [Block and Ochsendorf, 2007], which combines and extends graphic statics methods for application on shell structures. The features of the TNA method has been implemented computationally which enables an intuitive interaction with the boundary conditions, whilst still assuring compression only structures, a requirement for unreinforced masonry.

Evolutionary Structural Optimisation

A method that has proved popular recently is Evolutionary Structural Optimisation (ESO) which has been developed by Xie [Xie and Steven, 1993] and Sasaki

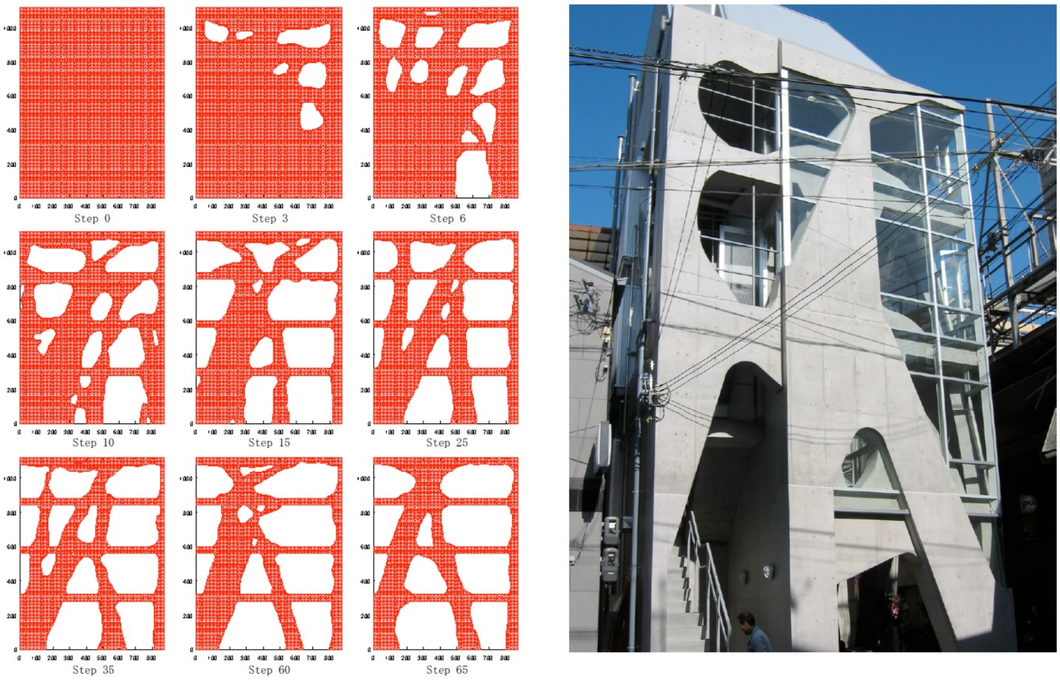


Figure 3-13: Example of early built application of ESO in two dimensions to design structure for building completed in 2005 by F-tai Architect [Ohmori, 2011].

[Sasaki et al., 2007] amongst others. Here the position of structural material is determined by considering a design domain and representing it as a volume of material with uniform stiffness. The desired loading and boundary conditions are then applied and a FE mesh is generated in design domain. After analysis is undertaken in regions where there is high stress, the volume is given higher stiffness and in regions with low stresses the stiffness is reduced. This analysis modification loop is then repeated with the stiffness increasing or decreasing accumulatively over the iterations until a steady state is reached. In this way, load paths are reinforced by defining stiffer areas of the volume with unused areas atrophying. The resultant volume with its variable stiffness is then converted into a built volume using an iso-surface that encloses part of the design domain, based on a volumetric stiffness cut-off.

This has many parallels with bone growth [Wolff et al., 1986] or tree development [Mattheck, 1998] and is a process known biologically as mechanotransduction. This has proven to reproduce optimal configurations for known boundaries such as Michel trusses. However, the real benefit of applying such systems is for un-typical design domains and load/support configurations. Here, optimal and perhaps novel solutions to a problem can be found with a low amount of input by the designer. With whole software packages devoted to the application of this method such as Altairs's Opti Struct [Schramm et al., 1999], as well as more conceptual-stage orientated tools like Millipede [Kaijima and Michalatos, 2014] which is integrated into parametric modeller Grasshopper, where they allow direct manipulation and fast solving of the ESO.

A benefit of the approach is that it does not require any initial form from the designer. By analysing and modifying the volumetric representation much like pixels, a structure can be generated with any shape and topology allowed by the resolution of the elements used. This is especially advantageous when a good solution is unknown.

This method does however present issues as it is capable of producing forms that are difficult to construct, as was the case in the Qatar Convention Centre 'Sidra Trees' project where an optimised tree like roof supporting form was generated using ESO [Sasaki et al., 2007]. The optimisation applied assumed the material used to build the structure would be something solid such as concrete. However, during detail design undertaken by Buro Happold, it was found that this would be too massive and complex to fabricate. This required post-rationalisation, which was undertaken by SMART Team at Buro Happold whilst the author was a member. This re-rationalisation replaced the solid curved members for hollow hexagonal steel sections onto which cladding shaped like the original design was hung. Whilst effective and arguably beautiful, it is hardly the integrated solution that was proposed at concept stage, and highlights the care that must be taken to consider sensible assumptions when using such techniques and ensure that buildability is not overlooked.

Figure 3-14: Sidra Trees project by Arata Isozaki, showing initial desired image, and resultant optimised structural section in red, and aesthetic cladding in white, images from [Smith, 2007].

The author has previously applied and combined these techniques in the investigation of innovative concrete design. A system was developed using ESO in order to specify where the material saving voids could be placed in deep concrete structures. The stiffness density could be translated into a stiffness field by introducing holes of different sizes into the material to create a variable voids ratio, rather than the typical ESO a solid/void iso-surface. What was required was a system for designers to interactively change the design and see what the outcome would be.

This solution was novel in that it was intended to be interactive and give users real-time dynamic feedback on both their design and the force flow created by an ESO type system. To have the system interactive without the need for large time-consuming FEM matrix solving at every step, a hybrid dynamic relaxation based solver was introduced alongside the conventional matrix methods. This method exploited the fact that the change in stiffness by the ESO process is incremental. As such, the deflection of the nodes will be close to what they were at the previous iteration, by implementing a 2D rectangular element each corner of which had two degrees of freedom, these could be used to calculate the updated deflections and voxel stresses with the new stiffness determined by the ESO. The DR system being linear in time to solve for a number of elements and easily parallelisable. When combined with a FE solution every 10th loop, the accuracy is maintained but not at the expense of speed. Applying DR to such a problem also allows for a more interactive experience where the designers can change the nodal fixity points and loading, and dynamically see what would happen to the volume's stiffness, by observing the holes resize. An example of this interface is shown in fig 3-15 and the pseudo code is below:

```
Input boundary conditions and load points
Grid domain
Build spring system
Build stiffness matrix
Set all domain elements stiffness to 0.5
Matrix solve to find initial position
```

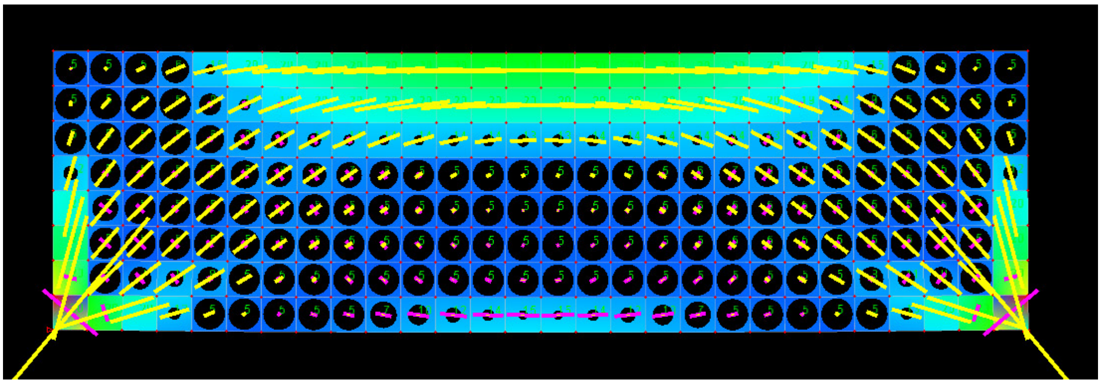


Figure 3-15: Image showing an ESO designed beam using the interactive system, note void being placed in the web of the beam. With stresses represented by element colour overlaid with principal stress directions. Source author.

```
loop = 0
While loop < maxNumLoops:
    Calculate element utilisation
    for each element:
        Calculate element utilisation (in basic case element average vo
    if element under utilised:
        reduce element stiffness
    if element over utilised:
        increase element stiffness
    if no change to any element stiffness or loops > maxNumLoops:
        break
    if loop mod numDRLoops == 0:
        matrix method solve
    else:
        dynamic relaxation method solve

    Update void distribution based on element stiffness

    if System KE < convergenceKE:
        break

    loop = loop + 1

Return optimised volume
```

Innovating with Structural Methods

In some instances, these techniques can be combined, innovated on or re-appropriated for a specific functional or aesthetic goal. One such example being a Smart Geometry conference workshop cluster organised by the author. In this instance, participants were challenged to design a minimum weight MDF cantilever which could take the heaviest load. Using a specific dimension of material and fixing and loading points, these were made using the large bed CNC cutter or milling machine. Support was given to participants in the form of example programs and code for ESO and Dynamic Relaxation, as well as support in developing algorithms and processes to link geometry and structural analysis. What resulted were new creative solutions along similar themes as shown in figure 3-16.

3.1.4 Construction Rationalisation

The author has collaborated with others on methods to model efficient structures for specific material and construction requirements, such as cardboard and other foldable sheet materials like plastic and metal [Maleczek et al., 2013]. It exploits the constraints of folding systems, to resolve the geometric organisation of reticulated shells. This presents a way to construct a network of elements but in such a way that the individual 'beams' made of a folded sheet that interface and connect well with one another, but also there is enough depth in the elements to resist the bending created by loading such a structure. This geometric solution was developed as part of Grasshopper, allowing it to be applied with relative ease to a large range of meshes, and allowing for recalculation in a reasonable time with any change in the mesh This parametric integration enables direct visualisation of 3D 'as-built' geometry along with the actual cutting patterns.

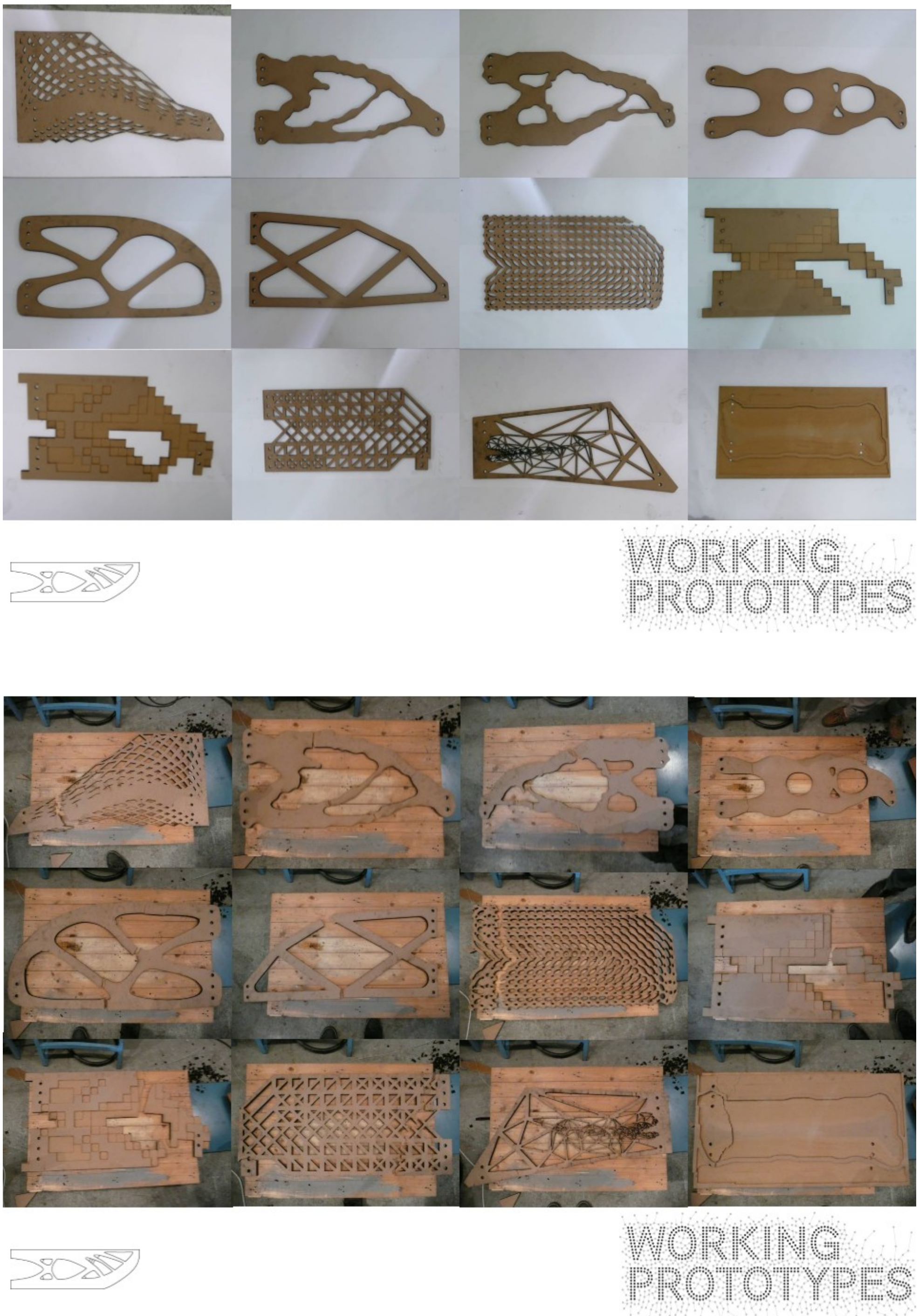


Figure 3-16: Image showing models before and after teasing of optimised beam design at Smart Geometry workshop cluster run by the author. Source author.

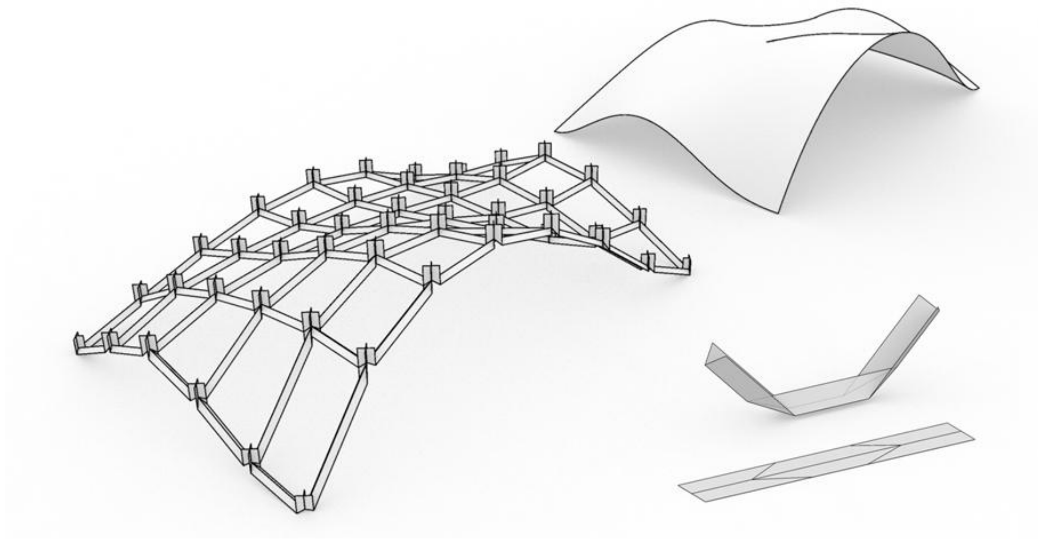


Figure 3-17: Example of grid-shell geometry generated from the base surface shown, elements are made up of simple rectangular strips also shown with only five folds required from [Maleczek et al., 2013]

3.1.5 Applications of Structural Rationalisation in Foster + Partners

Stadium Roof

There a are number of cases of applying bespoke custom solvers to problems at Foster + Partners. One example was in a competition entry for development for a stadium in Paris. The design proposed having two large sliding roof components covering the pitch, each being 55m wide and spanning 240m onto roller supports. A space frame was proposed by the lead engineer as the only way a structurally isolated element could span these distances. After initial environmental analysis of a basic truss, it was indicated that the pitch was in danger of over-shading the turf due to the truss. The space frame concept was further developed and it was proposed to look into an option, which used many thin elements, at tt was hoped that this approach would not create as significant shadows as a similar 'thick' option. This was due to diffraction of the shadows from the distance of the roof to

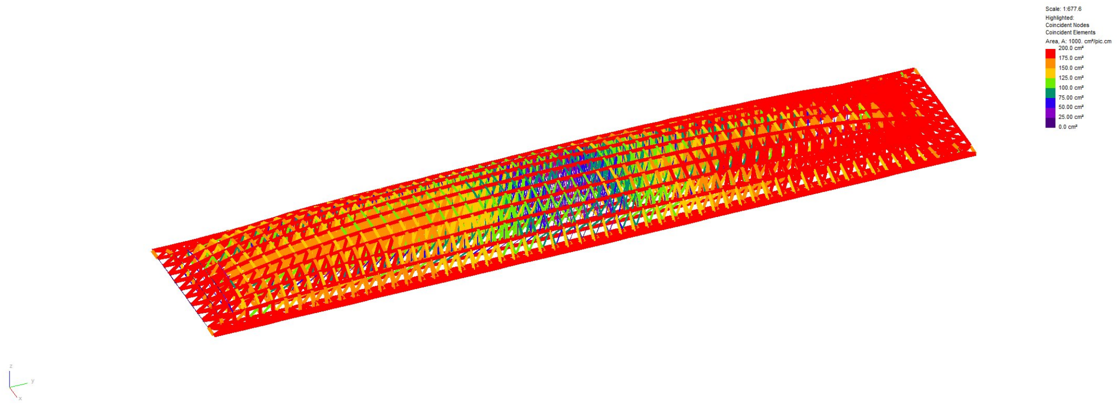


Figure 3-18: Image of one section of the roof, with the optimised section sizes range shown as a colour plot. Source Author.

the pitch, especially under diffuse light conditions.

This was the stage the author was brought into the project. The challenge was to create a system with minimal section sizes as possible. Instead of a single set of chords, a chord region was generated with many chords layers offset from the outside face and returning to the support points. The lower chords acting primarily in tension worked most effectively when straight. But the top layer required curvature and depth to promote shell and truss action respectively. A dynamic relaxation method was used to relax the top face with a uniform distributed loading along the length. , to provide an effective load take down for each face. With each of the chords was given a different load so it was as a different height and thus separated. All chords where then joined together to transfer the shear loads and make the space frame.

This space frame had a complex load path and there was no simple method to calculating element size, especially for deflection criteria which was the governing load case. So a novel custom methodology was developed. The core principle is the iterative sizing of a sections based on a solver with two criteria; element utilisation and deflection, or ultimate-limit-state (ULS) and serviceability-limit-state (SLS) respectively. This method extended previous work from the author [Joyce et al., 2011] whilst at Buro Happold and was applied on the Louvré in Abu

Dhabi [Shrubshall and Fisher, 2011], which itself is an extension of the work of Bill Baker [Baker, 1991].

SLS was taken into account by an iterative application of Bakers method of section sizing. This method relies on identifying a single point and direction from where to control the deflection. In this case, the selection was trivial being the centre span of the truss. Baker's method applies unit-load on the identified node in the direction for it to be controlled, and in a statically determinate structure, the forces in the structure are directly proportionate to their contribution in the deflection irrespective initial element stiffness, following the concept of virtual work. Thus, the cross-sectional area can be sized in proportion to these forces (also called deflection contribution) with a scalar value proportionate to minimise deflections as required. It is worth noting that this approach only works for determine load paths, as the load path needs to be independent of element stiffness which is not the case for indeterminate structures.

The extended approach attempts to circumvent this issue by applying Baker's method iteratively. Initially all elements are assigned uniform sections, which are then iteratively changed to create a load path that minimises deflections. For the ULS part of the problem sections are also increased or decreased based on the upper bound of each elements utilisation/max-stress after analysing with each load case. These two factors then contributed to the actual section size at each iteration, with this repeated until the section sizes became stable.

This process was initially achieved by creating a custom program to execute this logic from an existing GSA model. This was the fastest implementation of this method as GSA could be used to run the analysis and change section sizes and the model was captured in GSA for further analysis. However the initial form finding was carried out in Grasshopper, so it was desirable for this to be initiated from that environment. Extending functionality from the F+P Grasshopper structural tools, to set-up and return the results in Grasshopper without having to switch applications. The resultant method enabled elements to be uniquely sized from a large range of possible sections, but also enabled fast modifications to any part of process, both the relaxation or section sizing, which produced an efficient structure out of an already existing structural geometry and topology. Something that would have been a very labour intensive task and effectively impractical to do any other way.

This approach was used to understand the impact of gird size, truss depth and maximum allowable section size on the design. By limiting the section size under the same deflection criteria, one could trace the load paths by the increased areas on element density. Whilst the method was effective, it was found that the desire to have light diffuse around the structure was not achievable within reasonable parameters so the large door like roof option was ruled out. However, the tools developed were applied to other projects.

Mexico Airport

A project which has made extensive use of the Dynamic Relaxation process is the Mexico City New Airport design. The project is on-going and due to the commercial sensitivity not much detail can be revealed at this stage. However, the basic concept was to imagine one single large undivided roof that contains and unifies all the airport activities. To realise this the roof required exceptionally large spans to be achieved with minimal material. This was in order that the roof is minimally invasive to the flow of passengers and other program. To realise this goal it was identified early on that this would require flexibility in the definition of the roof, as the column positions would have to respond to the requirements below and be updated on a frequent basis.

The size of the column grid was exceptionally large, over 100m in many sections, with the overall building filling a 0.6 by 1.6km rectangle in plan. A comparison of this against a portion of central London can be seen in figure 3-20. To realise

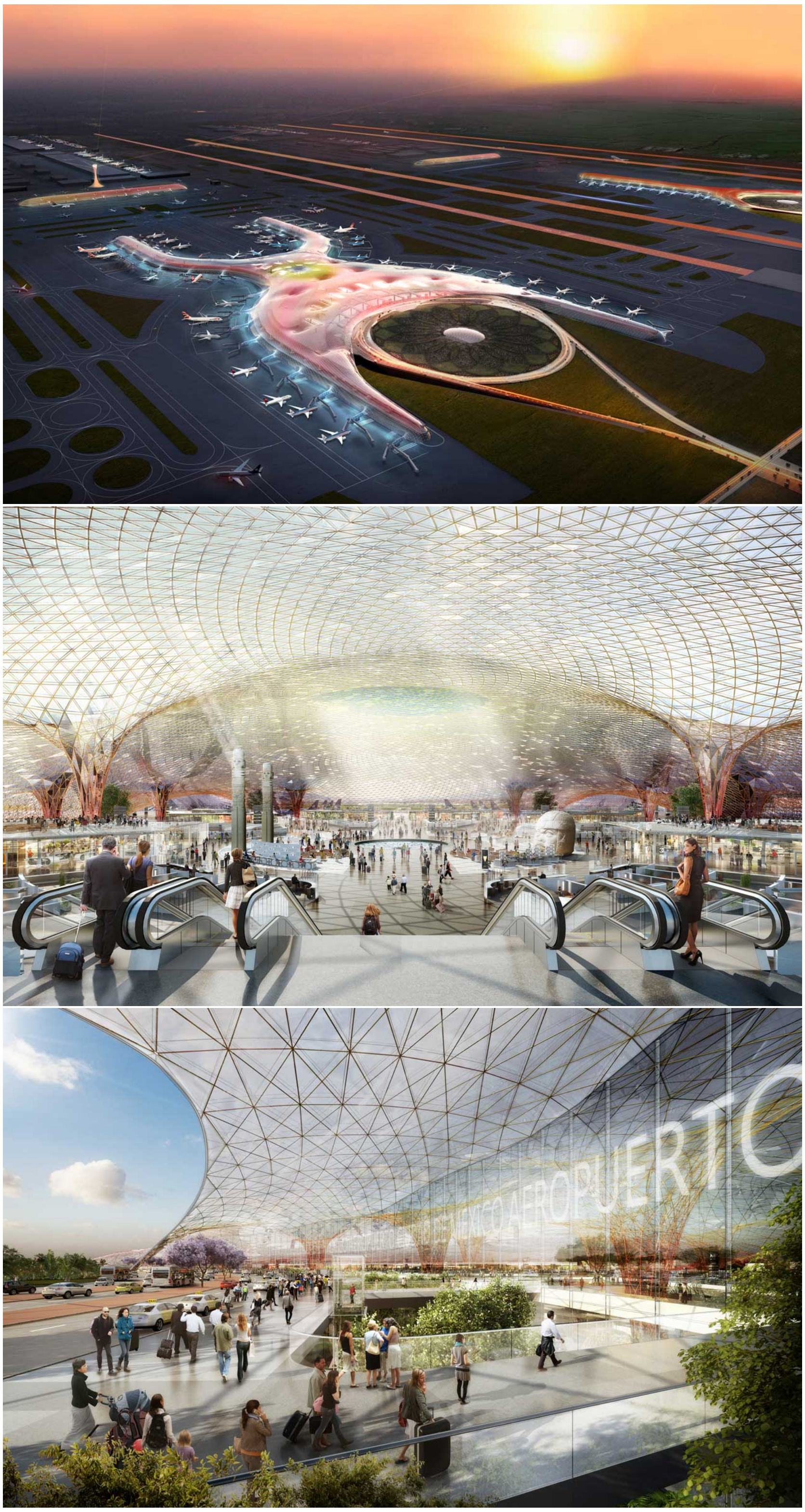


Figure 3-19: Renderings of the Mexico City Airport design showing the large singular roof and form-found columns. Source Foster + Partners

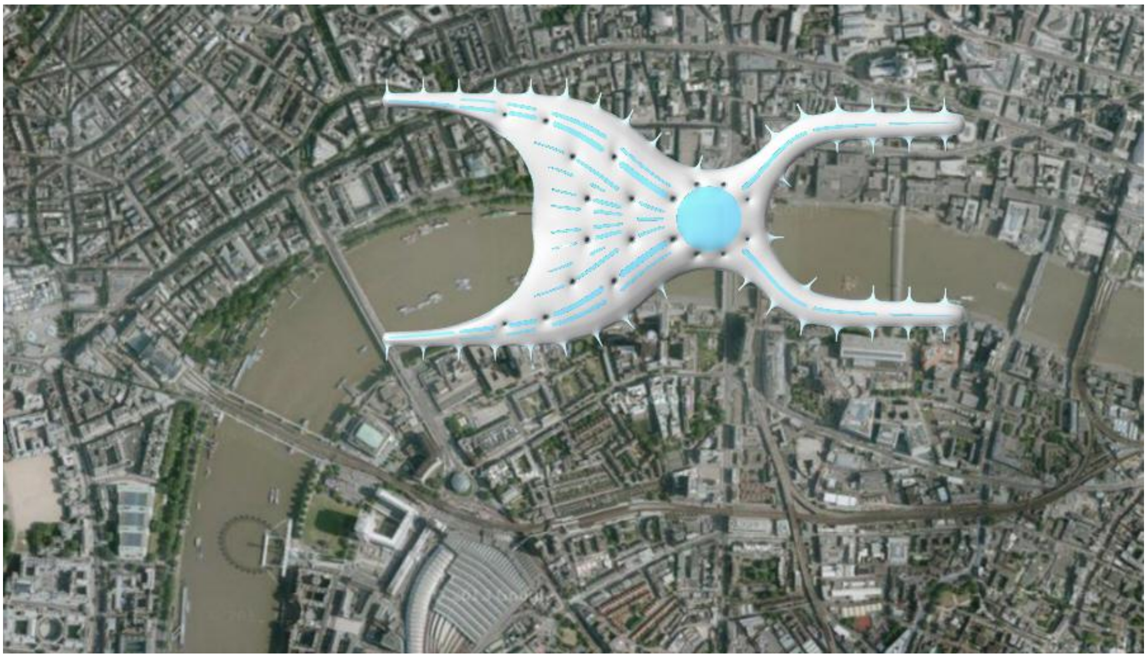


Figure 3-20: Mexico Airport design superimposed on London. Source Author.

such a structure, the use of dynamic relaxation to form find the roof was proposed. This was carried out by a member of the ARD group and the author.

Relaxation was able to create forms which minimise the out of plane bending on the shallow space-frame shell, as well as being responsive to changes in geometry as and when required by the rest of the airport design team. A roof of this size with a fine mesh (of lengths between 1-2m) required a high number of elements, which slowed down the relaxation significantly. However, methods to relax a coarse definition, then subdivide among other techniques were developed to improve the modelling speed of the system. This was undertaken in collaboration with local space-frame fabricator Geometrica, integrating requirements from them of minimum curvatures, average length sizes, node valence and angles into the dynamic-relaxation system, so to form-find as accurate and as efficient form as possible from an early stage. This was all realised within Rhino and Grasshopper, using both custom dynamic relaxation routines and Kangaroo.

A physical hanging chain model was also produced during the development of

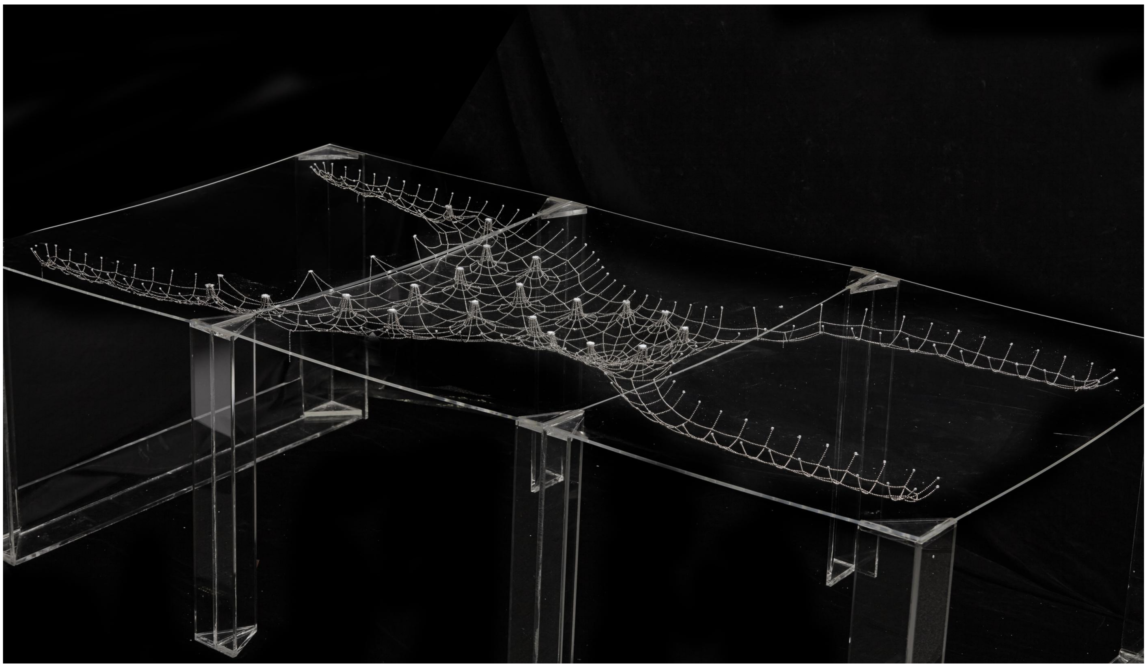


Figure 3-21: Physical model of Mexico Airport roof, using a smaller 'skeleton' set of hanging chains to outline overall roof form. Source Foster + Partners.

this project. However, unlike historical applications of these methods, this model was not used in defining the form, but rather made after the fact to enable those not involved in or familiar with the computational form-finding process to better understand and appreciate the behaviour of a dynamically relaxed system. It was also found that this model helped to sell the idea and principle to stakeholders as well.

This large project is currently ongoing and the author was involved in the creation of the competition winning scheme as well as design development of the project. Initially, the author was developing the structural concept as well as working on the form-finding and geometrical post-processing of the roof, but this involvement has broadened to analysis and functional improvement of the cladding grid and supporting space frame structure. This includes using the Foster Hub tools to build analysis models and return utilisation metrics as well as summation of measures of space frame constructibility. This has involved cycles of relaxation and structural and geometric analysis. It represent a significant part of the authors efforts in the practice, however owing to the ongoing design effort and commercial sensitivity no further detail can be provided here.

3.1.6 Discussion

This section has attempted to show how a variety of techniques have been developed which solve specific performance requirements of a design, both in literature and in practice by the author. They highlight the range of problems encountered and the novel innovations, which are often nature inspired. These methods show how with the correct understanding of what drives the performance of the design, the design can be led by intuitive design friendly systems. Furthermore it is possible to effectively apply these optimisations sequentially, therefore creating interesting designs with their own system of working and aesthetic. However care must be employed in their use as in applied wrongly this can lead to inefficiencies in construction as was the case with the Sidra trees project. As such there are good reasons to consider more general methods of solving problems; ones which do not require novel solution methodologies for every new instance, but also which are more accepting of initial constraints.

3.2 Optimisation

Optimisation strategies are differentiated from the previous performance based methods, as they have been developed to be independent of the specific problem they are applied to. There has been a trend in many fields of optimisation research towards greater abstraction between problems and solvers. This allows them to be used on many problems maximising development impact and minimising set-up effort. These methods have found to be growing in use in engineering and to a lesser extent architecture. The author has applied such methods to pertinent de-

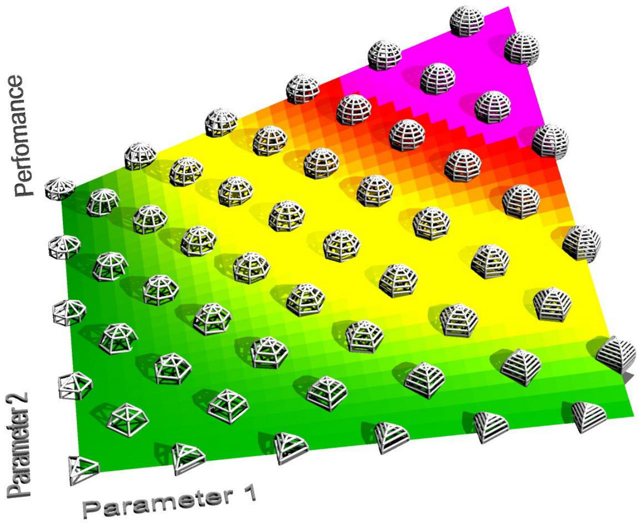


Figure 3-22: Example performance surface comparing two parameters of a simple model against a performance heuristic. Each position on the 2D parameter space represents a different model as well as a related performance value. Source Author.

sign problems. Specifically in cases where there are complex interaction of factors, or the previous methods have not been adequately developed to efficiently find good solutions for these problems.

3.2.1 Integrating Solvers

One of the most basic general solving method used in parametric design is called the 'Goal-Seeker' design pattern by Woodbury [Woodbury, 2010]. In this context a design pattern represents an atomic level of computational logic that can be applied to a design. The method modifies an input number based on a target output value of the design. This approach is essentially Newton's root finding approach, where the input number is iteratively changed to become successively closer to the desired but unknown input required for the known target output. The most basic form of this evolution is:

$$X_{n+1} = X_n - \frac{f(X_n)}{f'(X_n)}$$

As it relies on the derivatives/gradient of the function to inform the next move, these methods are often referred to as hill climbing methods. Whilst the solver logic is decoupled from the problem, there are also issues with its use. The algorithms success can find local optima rather than global optima if they exist in a system (it has multiple peaks). This is true in instances where the starting point is not close to the optima and thus will climb to the local optima. Other issues arise when the solution is a very small and discontinuous part of the performance function, or if the input values not continuous and differentiable there is no hill to climb. However despite this limitation it is often acceptable and useful in many design cases as an initial guess is already known, and has been applied to real projects [Dritsas, 2012].

These methods have been successfully extended for architecture and design by systems such as Killian's investigations into bi-directional solvers [Kilian, 2006], where the 'target' and the 'input' are able to be reversed during live interaction with a design session. This reversal of input and output to understand and 'tune' a design is typical in the design process. This approach has been extended and generalised even further with the work of Coenders [Coenders, 2012a]. Here, a network of relations is determined in much the same way as a parametric associative graph. However, breaking from the directed nature of those graphs, the system allows for nodes to be set as variables or fixed values in an arbitrary manner. The edges of the graphs explicitly encode relationships between nodes provided by the user. This definition of relations can then be set to solve a specific state of a system by determining or minimising/maximising unknown values.

3.2.2 Meta-Heuristics

Methods that react directly to the target objective can be effective however they often suffer from being too hard coded to the problem they intend to solve; Requiring an understanding of the relationship between input variables of the design and

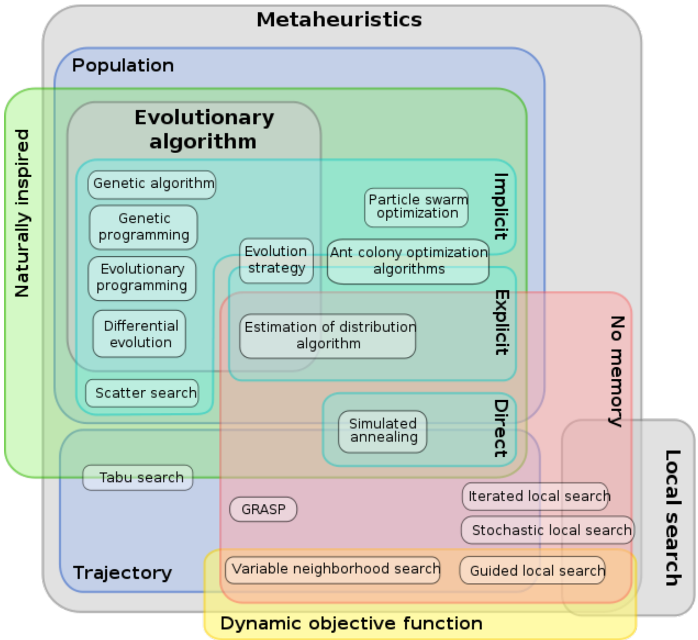


Figure 3-23: Example classification of metaheuristic search and optimisation methods from [Dréo, 2011].

the resultant effect on the performance. This can be problematic in instances where there are numerous inputs to control and their relationship is not well understood with respect to the desired outputs. This is especially true in cases when some parameters have little or no effect on the design, or the interaction of various inputs interfere with one another. This is often the case in large complex engineering and design projects.

A class of optimisation methods have come into use by the engineering community, called Meta-Heuristics, which attempt to address these issues. These solvers work on the basis of a separation between inputs and outputs, requiring the user to determine the function which represents their ideal requirements in terms of high-level properties or values. This is often formulated as a minimisation function, which can be the sum of different values to be minimised as well as the inverse of values to be maximised.

Solvers that are actively being employed for the engineering and architecture field often come from a class of stochastic solvers. These do not rely on being encoded with any prior knowledge or understanding of the behaviour of the problem, instead depending on numerous samples of the problem function. This higher-level approach, is what give this class its 'meta' name.

There is a wide range of solvers, each with their own advantages and disadvantages. The solvers, often use a physical or natural processes as inspiration such as Ant-Colony optimisation or simulated annealing. There are some reoccurring features of these algorithms; They typically initialise input values at random, normally more than one data-point at a time. Then, over successive iterations effective values are reinforced until the algorithm is only exploring productive areas of the design space. Finally, after a period of time either determined by convergence criteria or allowable run-time, the best result is returned.

The most widely applied by the author is the 'Genetic Algorithm'. This mimics the evolutionary process first explained by Charles Darwin [Darwin, 1871], and summarised by Herbert Spencer as "Survival of the fittest" [Spencer, 1896]. Originally proposed by Alan Turing [Turing, 1950] and popularised by Holland [Holland, 1975], it conceptualises evolution for the purposes of optimisation, with input values taken as 'genes' and the resultant design the 'phenotype', with the minimisation function being referred to as the 'fitness-function'. To start the algorithm, an initial gene-pool is created by random, genes can be 'real-coded' as a list of the direct input values such as scalar or integer input. Alternatively a binary string can be used and is more similar to genetic DNA code. This is then mapped into input values where it is converted to phenotypes. Based on the fitness of each individual, phenotype's genes have a higher or lower chance to remain in the pool for the next iteration. With the surviving genes being combined with other genes as well as being mutated by adding random numbers/bits to the genes, typically this results in each generation being successively better performing than the next.

Figure 3-24: An overview of the Bangalore Residential Project. Source Foster + Partners.

Optimisation Applications at Foster + Partners

This class of solver has been applied to active projects within Foster + Partners. In one exemplar case study, the Bangalore residential complex, these concepts have been explored most thoroughly [Tsigkari et al., 2013]. The project was relatively unique in Foster + Partners, there was little restriction or preconception about the general form of the design. Thus, it was proposed that the design be driven almost solely by the performance requirements. The overall aesthetic allowed for and actively encouraged a complex floor-plate which could vary between floors. It was decided by the design team including the author and ARD group to consider the building in the form of three-dimensional pixels called voxels. Then, to derive the logic to determine the off or on state of each of them, based on the functional requirements for the residents. It was decided in the climate of Bangalore that the key environmental aspirations would be: favourable winds, good views of both the sky and the surroundings and low levels of direct sun. These were translated into the respective quantitative analysis: insolation, vertical sky component, quality-

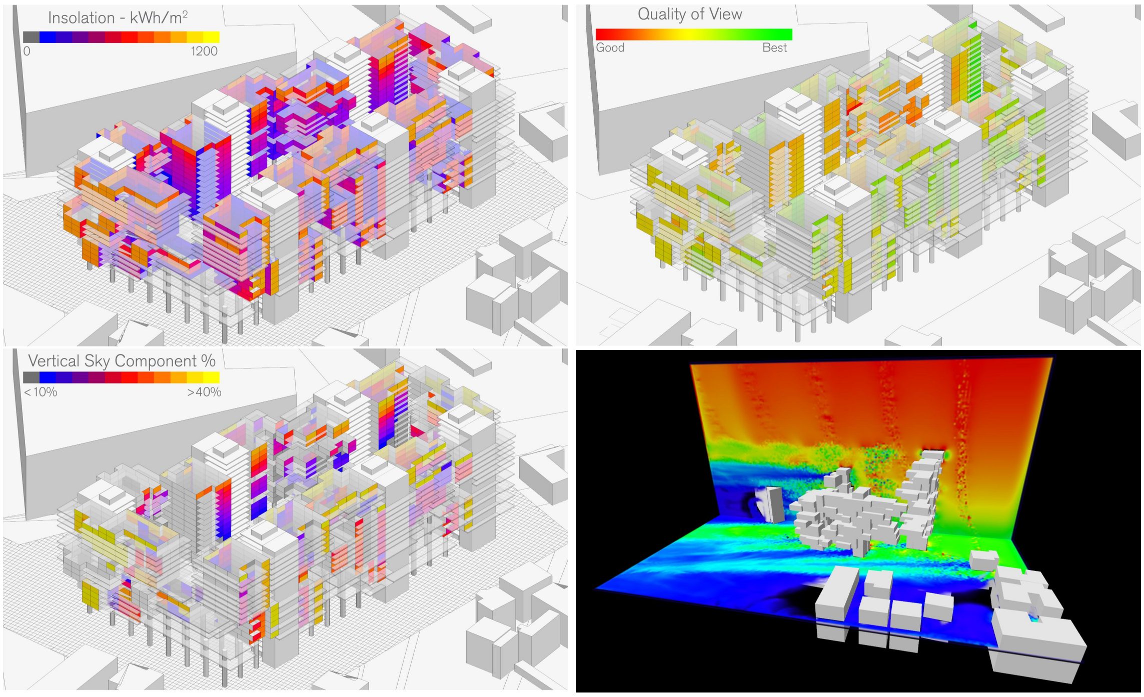


Figure 3-25: The four main driving analysis of the Bangalore voxel form. From top left; insolation, quality of view as calculated by [Davis et al., 2014], vertical sky component, and wind velocity as simulated by [Chronis et al., 2011]. Source [Tsigkari et al., 2013]

of-view and average wind velocity. This was done by integrating an internally developed solar isolation tool 'RadIO' by Martha Tsigkari [Chronis et al., 2012a], a fast fluid solver by Angelos Chronis [Chronis et al., 2011] and quality-of view analysis by Adam Davis [Davis et al., 2014], using a similar framework as identified in previous studies [Chronis et al., 2012b], with reasonable ranges chosen as a target for each criteria.

However the result, and more specifically the form of it introduced challenges for the realisation of an efficient structure. This irregular geometry required load-path in the plane of the shear resisting dividing walls whilst impeding residential the space as little as possible. The complexity of the load-paths from the floor plates through the walls to the vertical cores meant that a standard structural solution would not be an efficient solution. Therefore, a more adaptive approach was required to design the structure.

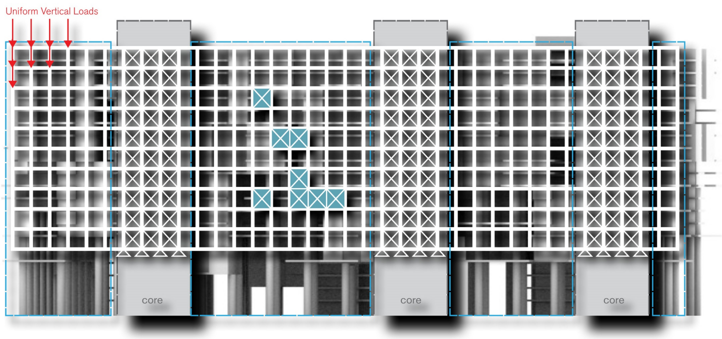


Figure 3-26: The initial structural frame configuration of cross-braced cores, other frames could them be filled in (or not) by the GA. Source Author.

A computational method was decided on, initially, based on the previously discussed section sizing method. This was applied to an indicative elevation slice as although a 3D version was possible it was felt that a better understanding of the optimisation result would be gained from observing a section. The slice was taken along the long elevation, which was considered the most complex load distribution, with three main core positions and four different spans. This section was then converted from 3D voxels to a 2D cross braced frame structure. At this early stage, the orthogonal elements of the structure represented walls and columns, with the diagonal elements representing steel bracing or concrete shear walls. This frame was optimised for section thickness using linear pinned elements, but assuming that this would relate to the reinforcing capacity of the concrete structure if/when the solution was examined more detail.

Although this method was effective in finding an optimal load path, it suffered from distributing the shear loads too evenly across the elevation, requiring too many braced walls, thus, preventing spatial connectivity between the two sides of the section slice. It was hoped that the result would have presented a minimal number of elements. However, this was not the returned result; essentially the optimisation was too locally concerned, and was not able to adapt to the global requirements of the structure. It was considered to introduce a penalisation factor to ensure some voids in the structure. For example, setting ele-

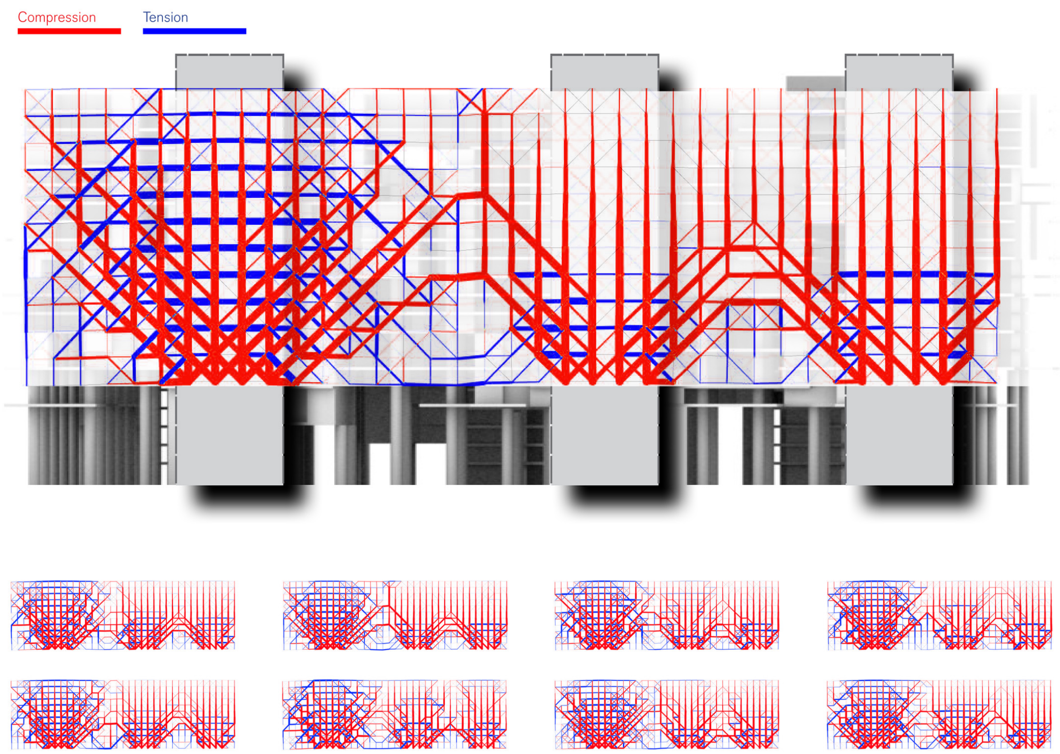


Figure 3-27: A utilisation based section sizing algorithm, this is then run by a GA (shown below) to determine the correct positioning based on material usage. Source Author.

ment stiffness to zero in cases that stresses are below a threshold. This approach is often employed in ESO problems to prevent unrealistic material distribution [Bendsøe and Sigmund, 1999].

However, it was identified that this might not have the flexibility needed. For example, whilst some porosity was required per floor it did not strictly need to be at any one place. This was a constraint that would be difficult to encode in this optimisation paradigm. Instead a more general genetic algorithm optimisation was employed over the top of the structural method to optimise the design constraints. This was chosen because of its relative ease in implementation of extra requirements as they arose, to further constrain the design [Goldberg, 1989].

To enable fast exploration of options, the study was implemented in the Processing environment , using a spring system to model the structural behaviour.

This light weight approach allowed for each iteration to be quickly calculated, meaning that an optimisation run could be completed in between 30 minutes and 1 hour. This was required as options for porosity and resultant weight and material distribution were explored by the author and the design team as a whole. The results allowed for a better creation and understanding of effective structural systems. However it was the ability to understand the sensitivity of structural weight to porosity that was most directly useful to the design decision making at that time. In the end it was decided that a core based structural system would require an unacceptable level of interference to program by shear walls required. Thus, a column and core supported version was eventually chosen, however the above work enabled this decision to be made.

This optimisation approach was taken explored by looking into all of the separate features of the design as one collective whole that can be optimised together. This was investigated as an extended study and captured in [Tsigkari et al., 2013]. Whilst it is true that with enough time the optimisation of quality-of-view, incident-insolation, wind-flow and structural efficiency can be joined together in a single optimisation, what was most problematic about this in a design environment was working out a comparable relationship between different objectives to compose an effective fitness function. An extra percentage point of vertical-sky-component does not have the same units as an extra meter per second of wind. This makes them harder to quantitatively compare. During the design process, it was proposed that these metrics could be condensed down to an economic measure. For example, good views although subjective, directly influence the objective value of an apartment value. Equally, it can be shown that good wind-flow and low insolation saves on mechanically assisted cooling. Although this was proposed it was hard to find a simple but reliable model to quantify the sum value of these components that the residents or developers would put on these factors.

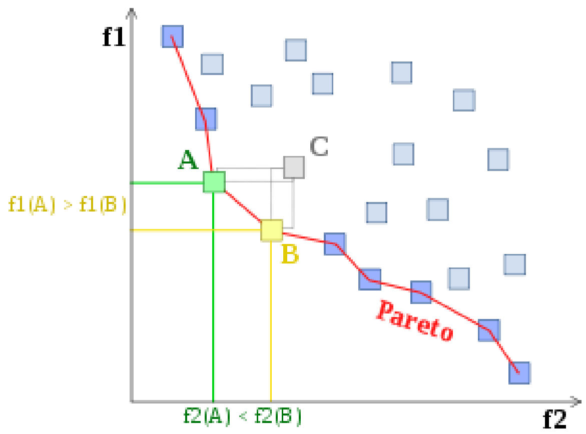


Figure 3-28: Example of a Pareto front in $\mathbb{R}^2$ objective space. In this example points A and B are non-dominated solutions as they do not have any points in their volumes and thus are on the Pareto front, unlike point C where both A and B represent an improvement in f1 and f2 with out any reduction in either value.

3.2.3 Multi-objective Optimisation

One solution to the problem of defining the trade-off between different performance metrics is by negating the need to explicitly define the trade-off ratios. It is possible to use optimisation strategies to uncover what trade-off ratios are existent in the underlying system. This approach was developed by Vilfredo Pareto and takes the form of a boundary referred to as the 'Pareto Front'. This front can be plotted in the performance criteria objective space. This boundary is a subset of all the possible designs, where each member of the set represents a design which is 'non-dominated' by any other. Non-dominated solutions are ones where there are no other alternative which offers an improvement in at least one objective without a deterioration in another other value thus is an optimal. This front can be visualised by presenting each option in objective space, a space equal to the number of dimensions as objectives considered. Assuming the goal is to minimise all values, an option is on the Pareto Front, if it can describe a rectilinear space from that point to the origin which contains no other feasible option.

The benefit of using this set being that rather than having to predetermine what the trade-off ratios are and only having one optimal value which represents the best solution for this, the front exposes all possible optimal solutions for any given trade-off ratio.

Whilst the concept of Pereto Fronts is relatively easy to understand, there is a difficulty in generating them for complex problems. Evolutionary solvers are well situated to these fronts as they rely on working with multiple samples of the design space and thus are able with modification to optimise for a set of non-dominated solutions (the front) rather than just a single optimum.

The most widely used of these is the Non-Dominated Sorting Genetic Algorithm or NSGA-II [Deb et al., 2002], although this is an area of active research. This method uses the same concept as a genetic algorithm. Relying on a heuristic which is still single valued, but in this case the fitness function has 'meta' Fterms that favours genotype-phenotypes which are non-dominated by the rest of their generation population. Additionally, there are terms which select for a well distributed set of solutions along the front, preventing bunching of similar values. Over a number of iterations the NSGA-II algorithm develops generations which are successively closer to the front. The resultant non-dominated solutions can then be used as the Pareto Front in a similar way to the optimal solution in a normal GA.

Multi-objective evolutionary solvers, although, more descriptive than single optimum solutions, are not without their issues. Depending on the number of points in a generation, and the number of cycles it may never perfectly match or describe the front. More individuals are required in a population to adequately trace the front. This is especially true with a large number of objectives. Thus, more points and cycles are required for a solution compared to a typical GA.

Beside the technical considerations, this method also presents challenges in being able to visualise and understand the trade-offs generated, especially with three or more objectives. With two dimensions a simple scatter plot can describe the relationships, but with three or more, there is a difficulty in visualising the results graphically, which acts as a barrier to understanding and interpreting the front.

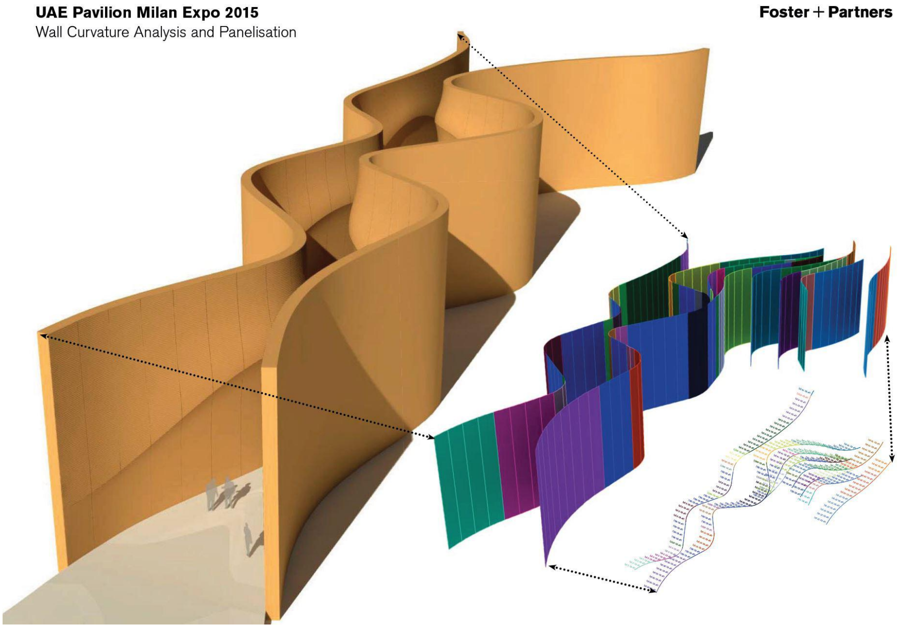


Figure 3-29: For constructibility the walls needed to be reduced to a series of repeating panels. Source Foster + Partners.

This approach has been shown to be very effective on a range of engineering problems. The author has worked with such methods in for truss design of a heliport [Evins et al., 2012]. In this case, a complex structural truss was generated using a parametric model. Due to the large spans involved, the deflection and material usage trade-off was important. In this instance, it was possible to generate a Pareto Front of this deflection material weight relationship. It was found that the relationship was not linear. Using this trade-off it was possible to see that with the previous deflection criteria, only heavy structural solutions could be found. However if greater deflections where permissible by careful detailing and cladding spec, then much smaller steel tonnages could be obtained.

Applications in Practice

Optimisation was used extensively during the rationalisation of the UAE Pavilion [Malm et al., 2015]. The process of converting the original complex geometry to a

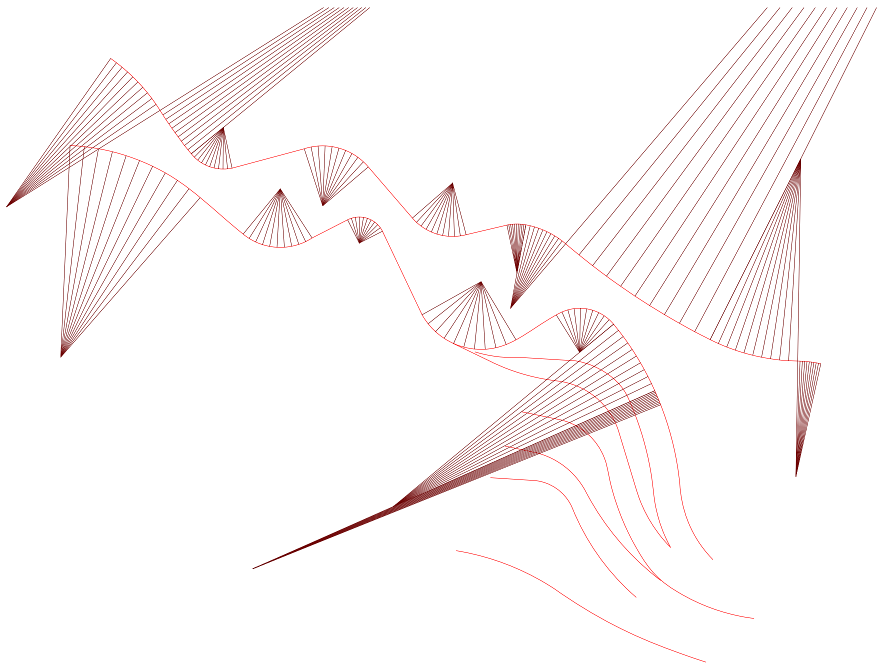


Figure 3-30: The UAE wall centreline geometry rationalised into arcs. Source Author.

finite number of build-able elements was critical to the building being delivered in time. There was a balance between the simplification of the geometry for construction purposes and over-simplifying and loosing the aesthetic appeal. These decisions would have to be taken at a number of scales and stages to get the result looking coherent. At each stage, optimisation was applied to maximise the aesthetic requirements against the functional performance and find the best compromises.

The earliest optimisations were converting the variable curvature wall centrelines based on b-spline definitions into a series of constant curvature arcs and straight line segments. Originally, this was attempted by hand, however, this soon became tedious and complex when considering all the wall curves. It was also felt by the author that this was unlikely to produce the most optimal result. To improve matters, Grasshopper's integrated genetic algorithm was used with the existing parametric definition. This used a series of arc start end and mid points which were chained to ensure minimum of $C1$ curvature continuity

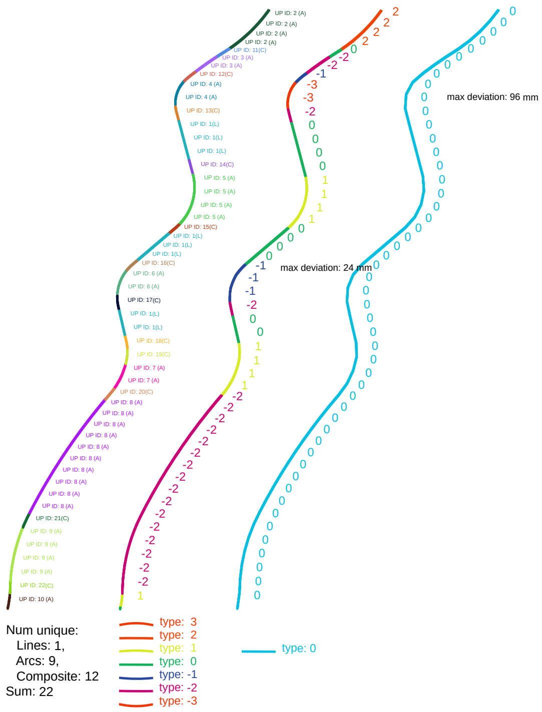


Figure 3-31: Study of segmentation of one long elevation of the pavilion wall for three options: arc-families with uniques, arc-families alone, straight panels only.

[Pottmann et al., 2007]. Then, deviation from the original was measured frequently along the length of the curve and the system optimised to minimise deviation.

Some modifications were required to improve the objective value by including the top set of deflected points, not just a single value, to coerce the optimisation to producing results where the optimum had numerous areas of bad performance, as only the minimum was being picked up. However these fixes produced high quality results with much less effort than the manual equivalent.

The second task required segmenting the walls into panels. Due to manufacturing constraints, all panels had to be the same length so despite the plan rationalisation arc panels could not be matched perfectly to plan arcs, which would have ensured that both their radius and length was the same. Different methods of discretization were investigated ranging from using; all the same flat panels,

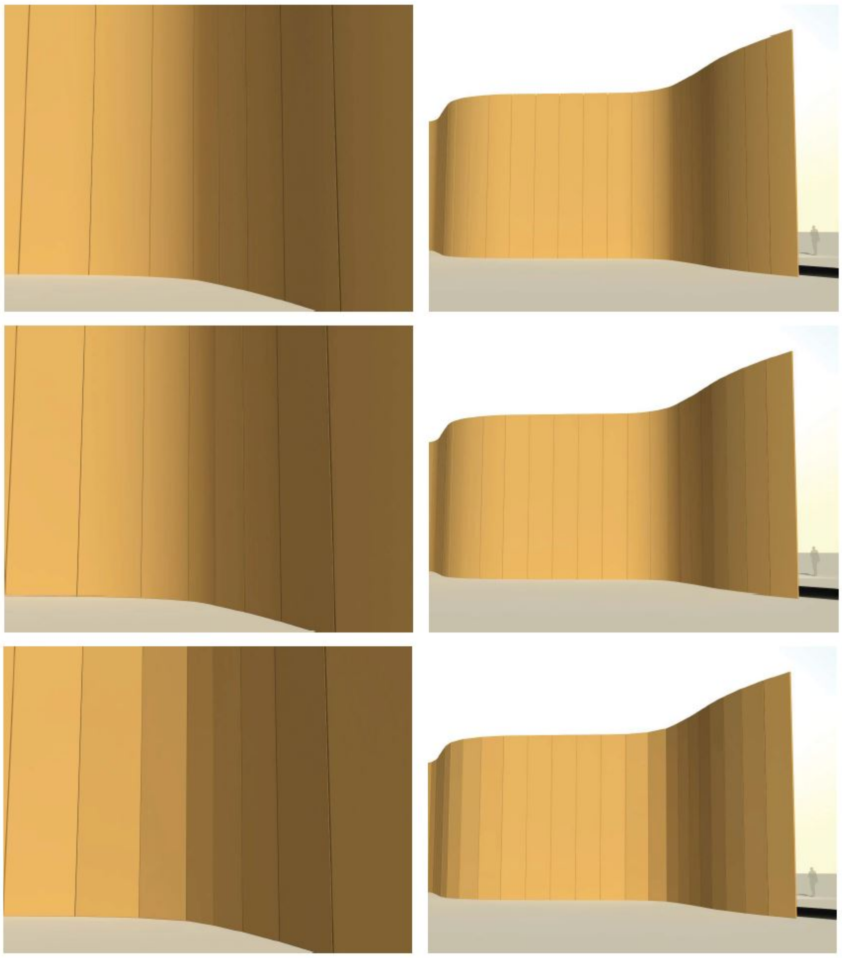


Figure 3-32: Study of panel options. From top to bottom: arc-families with uniques, arc-families alone, straight panels only.

restricted sets of arc-families and arc families with some unique panels at complex base arc transitions.

After visual analysis, including observing the panels in the augmented-reality Oculus Rift headset, it was found the angular deviation between the panels was key to the visual flow of the design. It was observed the simpler straight or basic arc-family options did not provide an acceptable result. There was concern over minimising the number of unique panels. So, a study was required to compare the number of panel moulds required for a given angular tolerance.

To accomplish this an optimisation was devised using a threshold value to dic-

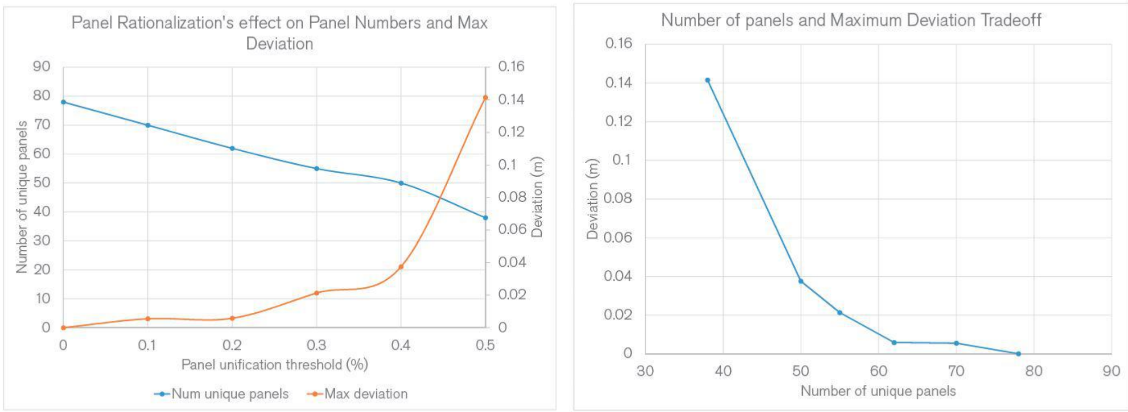


Figure 3-33: Graph showing panel deviation against number of panels trade-off.

tate when a panel needed to be made unique. In this case, the panel family arc-radius as well as an integer value for each panel was used to determine which family it belonged to. So, the optimisation was tasked with minimising the angular deviation between panels, a penalisation was also introduced to minimise the number of panels used. The system was optimised again using Galapagos with different runs for each threshold.

The results gave a a range of values for the threshold values, with lower threshold being more accurate but requiring more panels. This could then be tabulated as a trade-off between the maximum deviation and panel number. Based on the results, it was found that whilst using 50-panels had a large improvement over 30, any more than this had a diminishing rate of return and could be ignored. This understanding was more useful than having one sample as it allowed a relative comparison of the cost (in panels) and geometric quality.

A final optimisation effort was undertaken by the author's team into the pairing of the now decided panel arc families with the ripple types. Owing to the rigidity required from the moulds to keep the ripple geometry well defined it was not possible to use ripple liners to produce different ripple patterns easily for different panel arcs. Instead, the expensive liner would only be usable in one panel and thus arc.

To keep cost to a minimum, it was decided to minimise the number of liners to unique panel ripple combinations. So rather than the original 7 patterns which could be applied on all of the arcs, each arc-family need to use a subset of those patterns. Due to the fixed position of the arc-family panels along the wall, this impacted the proximity of ripple patterns to one another which would have made the pattern less obviously unique and natural than intended.

This was clearly a multi-objective problem, with a trade-off between the number of unique combinations and the number of adjacent or proximal ripple patterns of the same type. In this case, a function was devised which encapsulated the adjacency concerns of the design team. This gave an inverse exponential penalisation based on how many panels away the considered ripple type was from one of the same type. To elaborate, a panel with an adjacent panel of the same type might score 5.0, where as one which had the same type two away would score 2.0, three away would score 1.0, and panels with a copy more than 5 away scoring 0.0, as they where deemed too far to notice.

The number of panels and the adjacency function were used to set up a multi-objective optimisation. This used the ripple type for each panel as the genome. The plug-in Octopus was used for Grasshopper as it implements the NSGA-II algorithm and has useful plotting routines to visualise the data [Vier, 2013].

The results of the optimisation found that there was a relatively linear relationship between adjacency and the number of panels. Whilst the adjacency value was too abstract to understand, the resultant plots colouring close panels red and others green helped to make informed decisions about the preferred panel-ripple combinations.

The use of optimisation was very useful not only in getting good results but also saving time in doing, if not complex, then, at least human time intensive or 'fiddly' tasks, such as individually assigning over 170 individual panels. There where problems however, in defining the right fitness function; It was felt that the

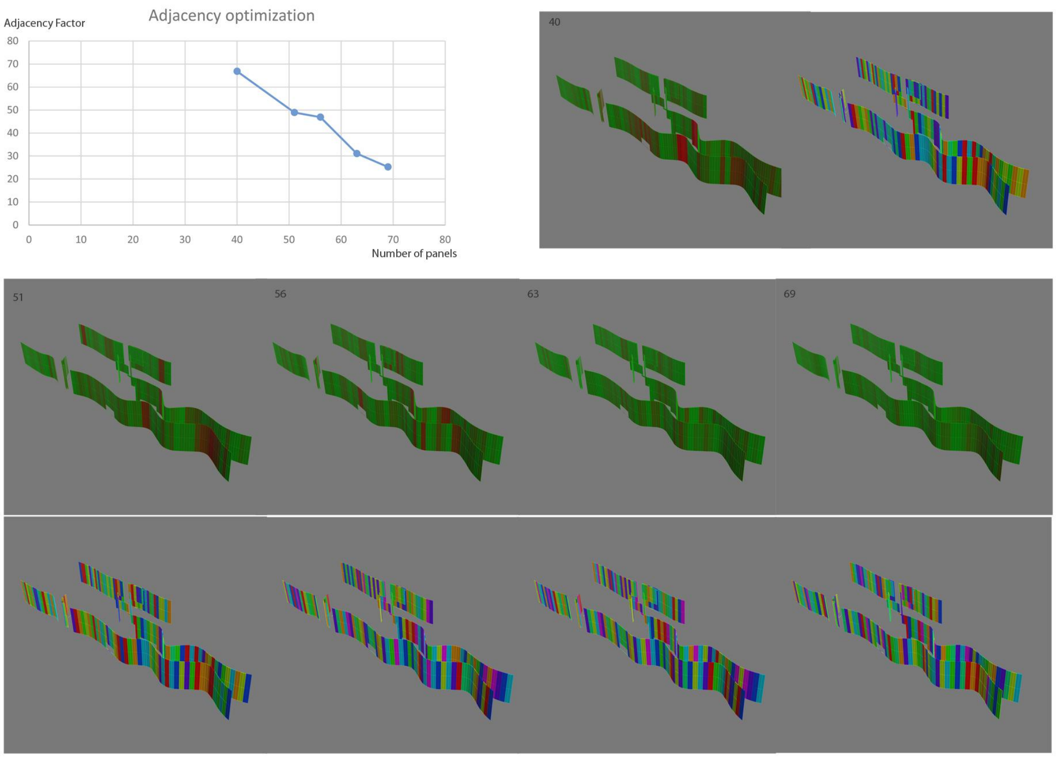


Figure 3-34: Results of the panel-ripple optimisation along with adjacency plots for each.

optimisation whilst effective, was always only partially representative of the construction and production issues faced by the contractors. Due to the incomplete understanding of their problems, the solutions presented were often countered by new issues in production. If it was possible to formulate the issues more comprehensively and quantitatively, then some of the rework effort would have been avoided.

3.3 Conclusions

This chapter has shown some of the ways that performance metrics can be used to drive the design. This can be done both directly in close-coupled systems such as dynamic relaxation, or in more abstracted higher-level ways such as with genetic algorithms. Whilst the more abstract methods enable greater flexibility and more objectives to be included, they can also stop design processes and obfuscate the relationships between inputs and outputs and their understanding. This can lead to a sense of powerlessness by designers and other team members associated with such a design process, even if they are keen to accept and engage in what an optimisation process produces.

A frequent problem stemmed from optimisation methods providing one immutable solution, which hinder further design exploration. Also, these methods have ineffective problem formulations, either due to the complexity of the problem or a lack of understanding on the part of the person generating the requirements. This can result in requiring numerous updates to the optimisation, leading to irritation and perhaps abandonment of these processes.

Despite this danger, optimisation is not only used to produce finial solutions. They certainly have their application in well understood problems at detail-design and construction stages to provide the best possible solution. However, there is arguably an equally important role in applying optimisation to understand and explore the design problem by finding good solutions from the start. This is an observed application of optimisation which has been termed 'Innovising' by Deb and Srinivasan [Deb and Srinivasan, 2006]. This is where an optimisation process is used as part of a human based exploration of the design space to explore and understand trade-offs, before making any concrete decisions on how to solve a problem, whilst exploring a problem and by observing what an optimisation routine returns and its success. A user might decide to further investigate or reject potential solutions.

This approach has also been observed in architectural context in a number of practices [Bradner et al., 2014]. With the availability of fast and easy to implement optimisation routines, a different approach can be taken to the use of optimisation. Typically, this can involve using optimisation experimentally to generate good first-guesses, helping to produce unimagined possibilities that by definition perform well. However as they are developed by algorithm, unhindered by preconditioned expectations, they can be widely different from human generated alternatives. It is this exploratory nature of applying such techniques that the author has found as being very useful along with the more typical use of improving a design by broadening the scope of considered designs in productive directions.

Chapter 4

Exploration and Understanding

> "Engineering problems are under-defined, there are many solutions, good, bad and indifferent. The art is to arrive at a good solution. This is a creative activity, involving imagination, intuition and deliberate choice."

Ove Arup (date unknown)

The previous chapter has shown some of the benefits of optimisation. However, this rarely represents the conclusive end of a design process, instead, often optimisation is present at the start of the design process to help guide intuition on what solutions should be perused. By 'innovising', design exploration can be improved by offering a range of high performing design options, obtained by applying meta-heuristic algorithms or special solvers to create plausible sets of designs as a instigator of design.

The application of such methods, however, put new demands on computer resources. This is especially true of evolutionary solvers, which by relying on many samples to optimise, require fast, numerous and ideally scalable model generation, all of which realistically requires automation in the analysis and data extraction.

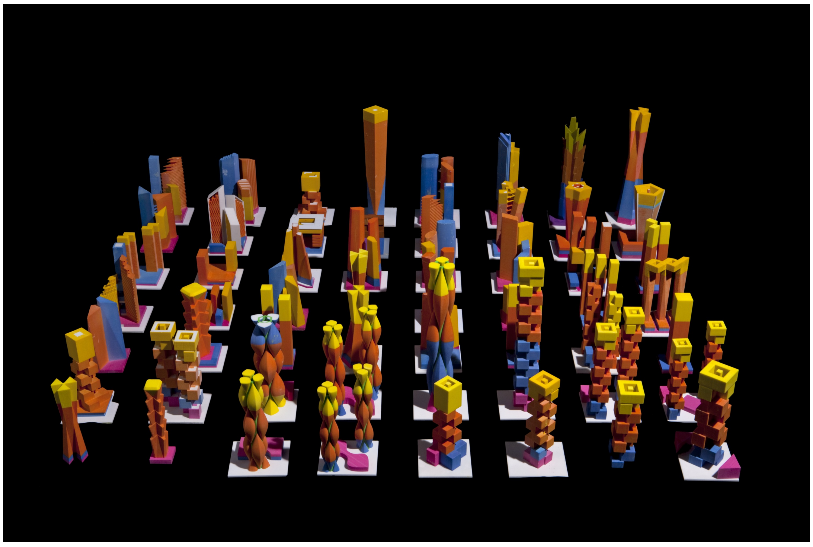


Figure 4-1: Example of the range of options developed for a project at Foster + Partners. Source Foster + Partners.

Furthermore, the design space explored by such approaches is fundamentally limited by the range of variability the optimisation is capable of producing, whether by parametric model or genotype to phenotype encoding. However most importantly the critical emphasis is on the user of such methods to understand and profit from the creation of such a wealth of knowledge. Thus deriving insight from data is a key aspect in this activity.

This chapter will look into approaches both in literature and applied in practice by the author to address these issues.

4.1 Exploration of the Parametric Model

In his essay "Design by Algorithm", Williams discusses the potential of describing the entirety of a designs definition with an analytical description [Williams, 2004], and this has great power for some forms. However, typically, processes are multi-stage involving moving data in and out of different CAD-environments, analysis software or breaking up logically separate parts of a computationally driven design. For example the griddling relaxation and non-linear structural analysis may exist as separate processes to a gridshell's form [Williams, 2001]. This modularity is pertinent for the focused development (especially if it is cross discipline) and reuse of a computational process. Typically, systems applied in practice are comprised of a series of parametric definitions which are linked manually. However, this raises issues for applications in large scale automated exploration of models.

One approach is to place the algorithmic definition of a single analytical definition of the geometry at the heart of the process. This is the design approach of the practice 'ijp' [Legendre, 2011], and this has created bridges such as Henderson Waves Bridge in Singapore. Unfortunately this is often not practical in large design process, where there is often different people working on different areas of the project, only combining scripts infrequently if at all. With singular monolithic algorithmic definitions which encompass the whole process, frequently becoming too complex, resulting in them being hard to maintain, modify or adapt to design changes as they occur [Davis et al., 2011].

What is more common in the authors experience, is the use of static output from one algorithmic processes being used as the input to others and so on. Arguably this modularity is a preferred strategy of development, but for optimization and similar end-to-end processes full automation of a design can be advantageous. Combining different scripts into more continuous work-flows data has not been explored in depth previously. As such this will be investigated in the first half of this chapter.

The ability to drive a algorithmic design model computationally allows for a different scale of information to be obtained from it. In previous studies this data has been used to run optimisations as well as explore trade-offs. However there is more understanding that can be gained from this data and approaches to display and explore this will be discussed in the second half of the chapter.

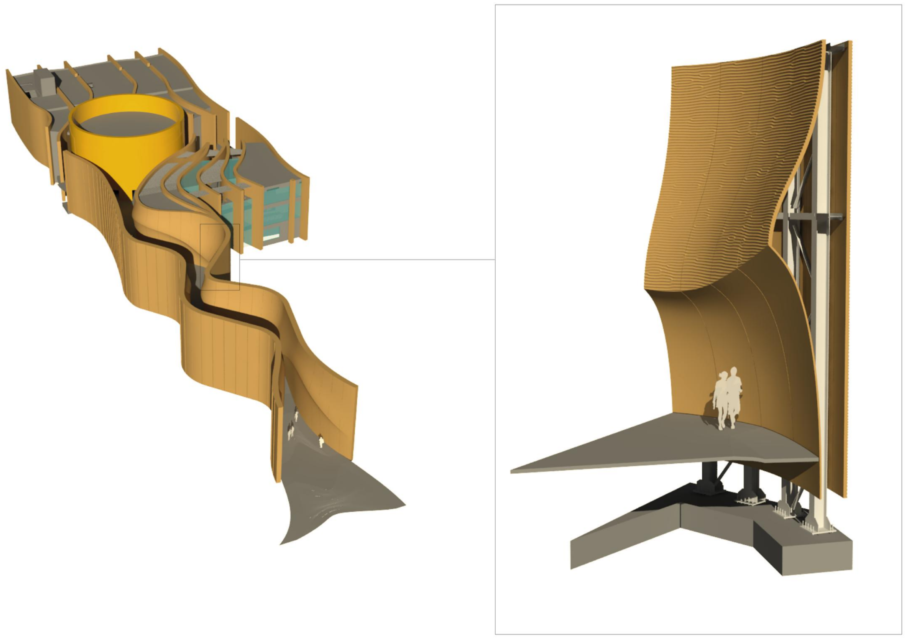


Figure 4-2: Detail of the UAE Pavilion canyon wall build-up showing individual panels interfacing with the structure. Source Foster + Partners.

4.1.1 Automation of Model Generation

One approach is to automate the use of the parametric software. In recent times a basic API has been developed which allows for the programmatic interaction and execution of Grasshopper scripts from outside of Grasshopper [Rutten, 2012a]. This allows for the creation of a meta-process above Grasshopper, manipulating inputs and parameter values, as well as storing and operating on outputs.

The first project that the author employed this approach, was for the mass generation of model data to realise the UAE Pavilion cladding panel construction geometry. As discussed earlier the aim of this project was to create panels that accurately evoked sand-dunes and ripples. Due to the novel ripple geometry, a high number of unique panels where proposed to achieve the desired visual quality. This was despite the rationalisation and in part due to the complex cutting. This consisted typically of a master mould making up the whole vertical elevation of

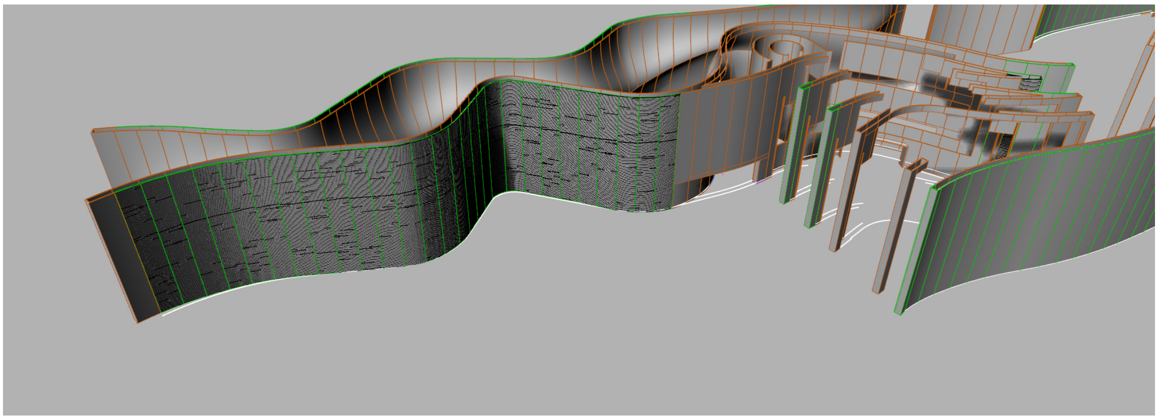


Figure 4-3: UAE Pavilion Rhino with ripple patterns applied to panels on one elevation.

a panel, along with two separate panel files which are the top and bottom, split along a ripple ridge. With the master mould being 1.8m by 12m high, with ripples 10cm in wave depth.

It was agreed by the contractor Can Build that full 3D Rhino geometry would be accepted as construction documentation. The documentation delivery required over 1,461 panels to be generated using a Grasshopper script. All GRC was modelled as complex volumetric poly-surfaces of B-Splines. To model each and every panel to construction tolerances for CNC cutting by the fabricator demanded a large amount of data to be generated, and would require a number of iterations of this process to perfect the geometry and hone the final constructed design. In the end over 6.66 GB of data was handed over, representing solely the GRC volumes, not including any other geometry such as support structure.

The patterns had already been determined by another script and the order of the panels was matched to the patterns and optimised so as to minimise visual repetition. The main process was therefore not logically complex, but due to the detail was still computationally intensive. A Grasshopper file was developed to create the panel volumes, which was used in design development and for the production of the mock-up. This assigned a ripple pattern from a limited family to a plan arc which located the position of the panel. These arcs were of differing radius and in some cases, length. This was extruded into a solid, trimming the edges that butted up against other panels. Finally, the panel was split into top and bottom panels following the unique ripple ridge line. The gross panel generation was encapsulated in Grasshopper with the panels splitting done in Rhino owing to an improved solid modelling capability.

Initially, each model (of three panels) took 20 minutes to compute. This was a concern as it was calculated by the author during delivery discussion meeting that this would lead to a 6.75 days running time, which was not acceptable on a project had a lead time of 2 weeks from design freeze, and considerable unavoidable manual modelling and checking down-stream. Thus, a way of speeding up the process was required at the risk of holding up the delivery of the project.

The GH code was optimised to improve running speed. A meta-process was developed so the manual cutting task could be automated via Rhino Script. This was extended to the creation of panels by automatically changing the input of the Grasshopper and 'baking' (exporting) the output ready to be cut. In this way, the initial 20 minutes was eventually improved to between 1 and 1.5 minutes a panel. However even this improvement presented a potential 8 hours plus runtime per iteration, and with the desired panel geometry being continuous between panels, any change would require a full re-run of the geometry.

The solution the author identified was to parallelise the task between multiple computers. A set of eight similar machines where specially built for the task, by the IT department; each cloned or 'computer imaged' from the author's machine, with each being accessible via a unique name on the local network via remote desktop. When an option was ready to be extended to all panels, all files were saved to the network. Then, each machine was initiated via a short command-line command, which opened up Rhino and ran a Rhino-Python script. This script opened up a read only base geometry Rhino file and accompanying Grasshopper file. Then, using a central network accessible manifest which contained all 'tasks', in this case

Figure 4-4: Eight identical Lenovo S20 machines networked as part of the F+P cluster. Source Author.

panels as rows in a comma-delimited text file, the script found a “not-done” task and set this to in-progress, using the aforementioned meta-process to generate and cut the defined panel. In the final version this took the panel id and the pattern id as inputs to the Grasshopper definition, to produce the panel at that location with the desired pattern. The script then automated saving the resultant geometry both master file and cut panels with the correct naming convention. Recording the panel as complete on the manifest before finding another unprocessed panel to do. Error handling was built into the meta-process to notify via the manifest of any tasks failing, before moving onto another.

This approach enabled a parallel non-blocking method of generating panels, which with a carefully implemented manifest writing handler is able to wait to read/write if other machine instances were using it, ensuring scalability to a large number of machines. For this task along with the 8 dedicated machines, any additional compatible computing resources could also be included, typically bringing the available machines to 10. All of these working on the problem resulted in a completion time of roughly 1.5 hours a run including set up, which was acceptable to implement and test changes. It was possible to have the manifest open and see the other remote machines update it to track progress of the overall process.

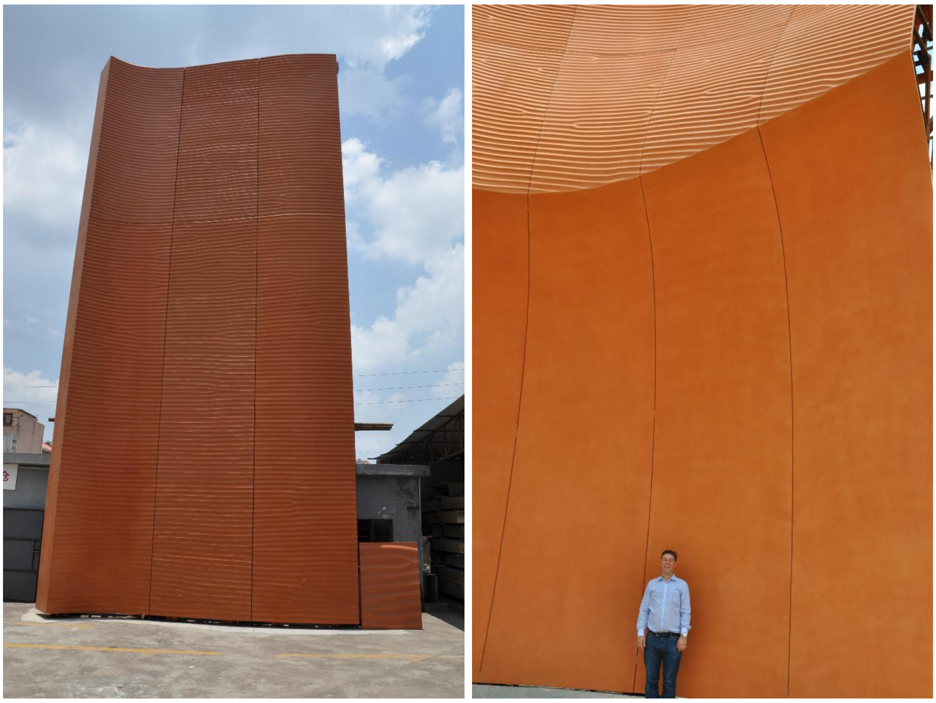


Figure 4-5: Mock up of exterior and canyon panels for the pavilion. Source Foster + Partners.

This cluster was very useful not only to deliver the model on time to the contractor, but also enabled more iterations to be undertaken to improve the design. It is estimated that generation of large parts or all of the panel design at construction quality was undertaken over 20 times over the duration of the project.

This allowed for the large scale generation of complex geometry, by scaling-up the initial process and automating it by embedding it within a meta-process. This 'task manager' script was written in such a way that any Grasshopper based task could be controlled, including pre-processing and post-processing any geometry via Rhinoscript, and has also been used on other tasks. The master meta-process has since been abstracted out as a function which takes the manifest variables as its arguments, and return the values used to fill in thee data in the manifest such as process success or verification/performance data, with the master process modified or rewritten upon each application to suit the requirements of the project.

There are limitations to this approach. Due to the parallel nature, the instances

Figure 4-6: CNC Machine cutting UAE Pavilion's sand ripple mould liners. Source Foster + Partners.

of the geometry can't interact with each other. All output geometry must be saved as a separate file as the base file must be read only. This means that aggregation maybe required to bring it all together. However, this is equally well suited to optimisation and generation of many options.

4.1.2 Computational Exploration of Design Space

Over the duration of the author's experience in working with optimisation, one of the key problems has been the abstract nature of the results and how they relate to the output optimums and why. Design is rarely a single one-shot exercise and often the most useful thing from a design study is not the actual design itself but a better understanding of the problem. Developing a feeling for the relationship between variable properties, and the resultant performance characteristics is invaluable as the design process progresses. Understanding the quantitative metrics along with qualitative appraisals and results in an understanding of the wider design domain, enabling those involved to steer the projects focus on the most promising areas.

The practical reality of obtaining this level of understanding for a specific project is not so easy. Experience is arguably the best route, if an individual has already completed many different similar projects, then, they are able to use this experi-

Figure 4-7: Installation of one elevation of panels. Source Foster + Partners.

ence to filter out the important and unimportant factors to focus effort on what will improve the project most. However architecture is very expansive in its reach and it is common for architects to be working on very different building typologies that they might not have experience of. This compounded by the fact that design projects often take years to complete making experience a hard fought commodity.

The alternative to directly knowing what is good for a design, is to experiment with different versions; This effectively builds experience, by allowing comparisons between options, leading to an understanding of what works or not. Producing many options, or 'Optioneering' as it is often called in Foster + Partners, is a frequently used approach to understand the design problems and solutions better by a team not familiar with the issues. However, this is very time and resource intensive, as it can require many different options to be produced and compared, often over many iterations, with the resultant understanding built up relatively slowly.

One approach that the author has employed to improve this situation is the auotmated production of larger ranges of parameter space. The basic idea being to lever the benefits of parametric modelling not in a retroactive way to comments etc, but in a proactive way, generating all options for consideration. Whilst this may sound very time consuming, as shown previously by employing developing meta processes on-top of the typical design work flows, this automated approach can be quite practical. With the resultant range of options enabling a faster less iterative overview and understanding of the design space.

The author has employed this on a number of projects, initially, for a relatively simple link bridge portion of a project for Tocumen Airport in Panama. The link bridge connected the old terminal with the one to be newly built. More accurately, this was not a bridge, but a moment frame with a regular grid of ground-baring columns which supported two floors; The lower floor for the top hung baggage transport and the upper floor for passengers in transit. The unique feature was due

Figure 4-8: Overview of Tocumen airport extension. The link bridge connects the existing terminal identifiable by the two satellite gates, and the new terminal on the left. Source Foster + Partners.

to the top hung luggage handling system; With the first floor height relatively high and heavily loaded, and large elements across the length of the bridge to support the load. This caused a very dynamic and lively structure, which had fundamental frequencies that were un-ideal for earthquake response spectra found in the region. Furthermore the first fundamental frequency, a translation in the horizontal plane was very close to the third fundamental frequency, a rotation or torsion of the top of the bridge relative to the bottom. This would result in a high chance of a combined swaying and rotation during earthquake, which would impart high moment and torsional stress respectively on the elements. This was unacceptable for a core linkage between the terminals so needed to be resolved.

The primary challenge with the structure laid in the fact that improving the existing dynamic response was not straightforward. Initially, the in house engineering team spent over three days manually tuning the structure using a trial-and-error approach. However, this did not result in an acceptable solution. They found if greater stiffness was introduced then this gave extra mass to the top of the structure and thus the frequency was made worse, by lowering the natural fre-

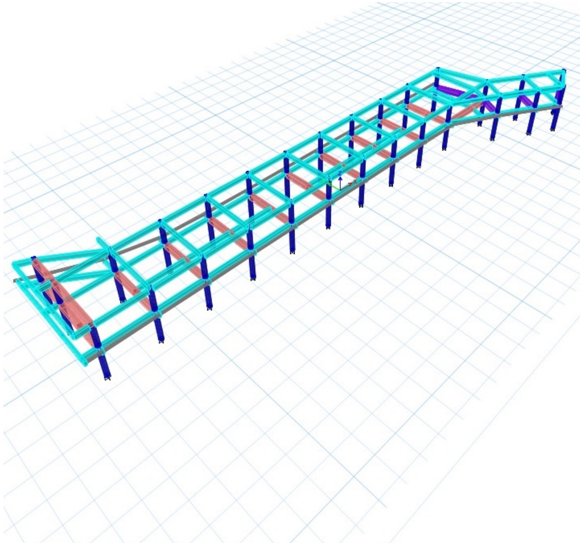


Figure 4-9: The structure for the Tocumen link bridge with each element group in a different colour.

quency closer to the earthquake frequency. There were three main beam groups used for this design, and complex interaction of these further complicated the selection process. At this stage an investigation by the ARD group and the author was discussed and initiated.

The approach taken by the author was to algorithmically calculate every combination; It was identified that the beam groups could reasonably consider nine different sections and still be suitable for other structural requirements. This resulted in 9^3 or 729 different options, which was considered reasonable for an automated approach but impractical to manually investigate fully.

A script that ran ETABS 2013 structural analysis software was written, using a different section combination at each iteration, with the same loadings and dynamic analysis tasks. After completion, the model and results of each where saved, before moving to the next model until all possible combinations has been generated.

In this case the script was run overnight. Based on an initial trial run, it was estimated that this was ample time to complete the task, with only one machine to

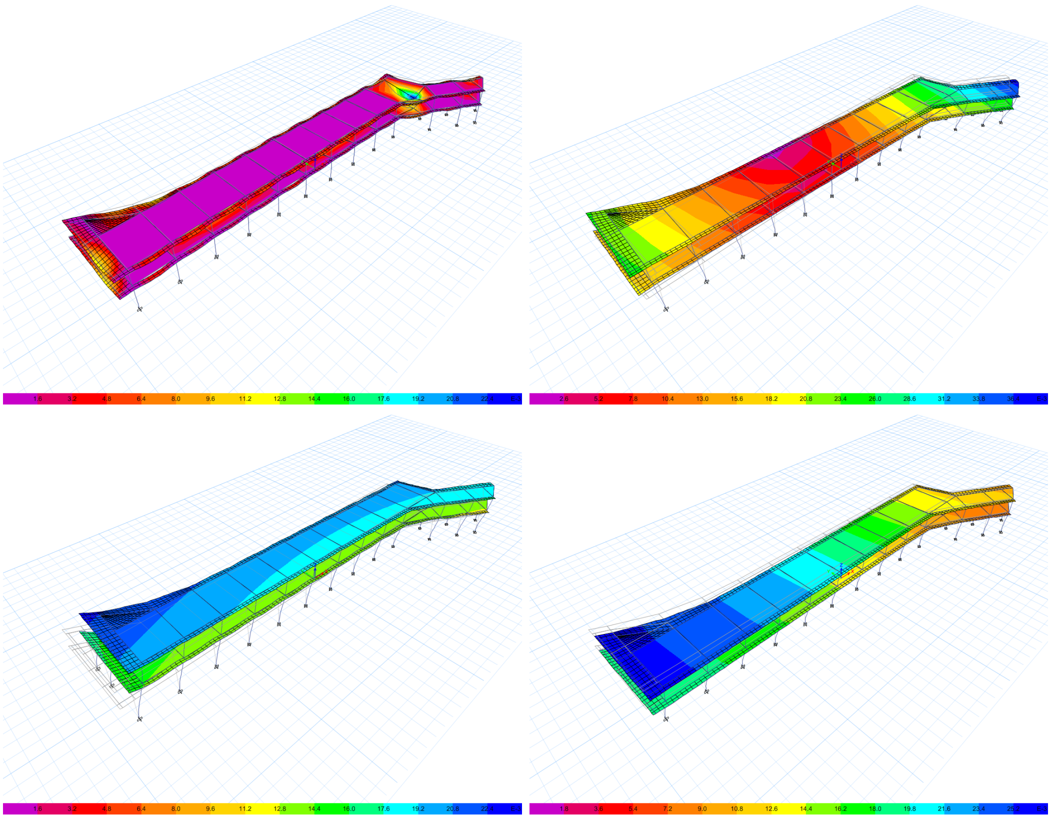


Figure 4-10: Link bridge deflection diagrams under in left to right; dead load and modes 1, 2 and 3 respectively. Floor plate colour is varied by deflection magnitude. Note the rotational mode 3, with a radial deflection plot.

Figure 4-11: Tocumen airport extension on site, with link bridge foundations and columns being constructed. Source Foster + Partners.

complete by early morning. By using only one machine it would reduce blocking access to valuable ETABS licences, that might be required by others.

After undertaking this process it was then possible to collate all the data and assess which was the best performing option for the desired criteria as all possible combinations had been exhausted. It was possible to plot out the Pareto optimal fronts for the various metric, and also compare the relationships between the change in one parameter and its effect on the objectives. Essentially finding what was significant to the behaviour of the structure. This helped the engineering team to understand the relationships between section choice and modal response. This study generated a much more thorough understanding bu the design team over the trial-and-error method which only presented a few models.

Whilst this arguably was a heavy handed methodology, the actual model creation and analysis was able to complete in under a minute per model. Thus it was feasible to run overnight, but could be further reduced by parallel processing. After the combinations have been created, the model and the results were saved, summary data was saved to CSV which was plotted and the results conveyed. During this study, a new metric was required to enable better decision making, in this case the concrete volume. In response it was possible for the author to write a script to query and automatically extract this data from the now pre-existing saved ETABS model in under an hour, and append this to the summary data. This would have been a similar length of time if much more data was extracted both model data and analysis results; As the opening a closing of each model required the most amount of processing time.

This approach represented a very useful aid to understanding this kind of engineering combinatorial problem and there was interest in creating a tool that any engineer could use. It was agreed there was a need to develop an excel plug-in that could generate and query a whole range of models based on a parent model and values to change, with the ability to then show the results as excel data. A proof of concept was developed, but owing to other project commitments this was not realised to a level of resolution for a fully functioning company wide tool.

Generalisation of Parallelism in Design Exploration

The previous example was effective for the problem it was trying to solve, however it was relatively limited, acting only on structural software. This process, however can be generalised and improved. In one case study, a more flexible approach was required which enables these brute-force methods to be generated more intelligently. This design problem was also another structural frame, but now with changes in geometry as well as sections. As such, Grasshopper was used as a platform on which to undertake the generation of the logic which would be necessary to undertake a design space search. By using the structural tools and carefully applying the replication or 'tree' features of the software, it was possible to generate all the models and return all the structural results as well as the geometric results such as material volume. The creation of all the models however was again linear, with Grasshopper making one version after another. One major issue with Grasshopper used in this way, if there was any error, then the whole set would have to be rerun or carefully modified, to exclude those already successfully processed, but typically knowledge of an issue would only be apparent at the end of a run. This issue aside, having a familiar geometric environment with many available plug-ins to speed up development and interact with other analysis platforms such as those providing structural or environmental metrics. This presented a more universal model for applying design search to parametric models. However, parametric systems such as Grasshopper are not developed with multiple model creation as a primary objective. Whilst they are able to cope with such problems using the same features that allow them to handle sets of geometry, a more robust, scalable and parallelizable system at the multiple model level would be an improvement.

Although the above approaches may not represent an ideal implementation, it is worth looking at these in the context of trends in computing in general. Moors's Law [1] is still regard as an accurate prediction, however, in recent times individual processor or clock speed of computers has been relatively stagnant. Instead, CPU chips have developed multiple-core technology. This allows them to process multiple instructions in parallel computing thread; with low-level software acting to aggregate the smaller computations and load balance these threads between the cores, with larger computing systems adopting 'cluster' architectures, whereby a network of slave machines are centrally controlled to solve large tasks. This has had advantages in preventing processors from 'freezing' by overloading single-core machines, and made large scale computer cheaper by using inexpensive commodity hardware to solve large computing tasks.

This change is still relatively recent with consumer dual-core chip architecture released in 2005 by Intel. As such, it has taken time to trickle-down to the rest of the ecosystem of operating systems and applicaiton programming interfaces (APIs) so that these compute resources can be utilised. These advancements have required new development strategies including programming libraries to fully

[1] Moor's Law is the prediction by Gordon Moore the co-founder of silicon chip maker Intel Corporation that every two years the number of transistors on integrated circuits such as computer processors and solid-state memory. This has a very direct impact on the speed of processing and amount of RAM in computer devices.

utilise the power available, and a departure from the classic singular processing paradigms which are still widely adopted by programmers. As such the application of these new approaches has been slow; A good example of this issue is engineering solvers, which whilst now implementing parallel sparse matrix solving routines they still have major sections which are not parallel. A typical bottle neck being preprocessing, which, based on the users experience with very large model generation (of over 600,000 elements) requires over half and hour to just check data, where the supposedly 'hard' solving is now complete in a fraction of that.

This slow rate of transition is true in CAD and parametric software, where the relatively small development budgets and resources put priority of working reliable behaviour over speed. However, there are major plans to parallelise both Grasshopper and Kangaroo in the next major releases of both [Rutten, 2014], [Piker, 2014].

As such the creation and processing of multiple models is something that looks likely to rise, but by being able to control these processes at a higher level via meta processes some of the reliance on external software to speed up can be reduced by scaling up the number of parallel instances being solved.

4.1.3 Broadening Design Search

In the previous section, we discussed the ability to explore the design space offered by a parametric model. While this can offer an essentially infinite range of options, provided by the continuous scalar parameter values of a parametric model, in reality this type of variation can only ever have so much impact on the design, due to the parametric model's limited flexibility. In essence, the range of possible designs is already bounded by the user, as they build and define the model and thus, the design space to explore. This is a very real and hard limit to what can be achieved in getting the best performance from a design.

However, there is really no reason why the parametric associative paradigm must be the only one prevailing. Indeed, there have been many varied examples in the world of computational design which proposed much more unbounded methods of design exploration. Some of these will be discussed as well as their application by the author.

Automated Design and Artificial Intelligence

The application of computers to design pre-dates the development of parametric modelling. Currently, there is an emphasis on the computational assistance of representation, by using parametric systems to capture the geometric construction of a design, then, allowing the parameters to be changed and thus, options are explored or tuned. This is still very much a wilful system where the major design decisions are taken by the person developing the parametric model.

Earlier to this period however, significant effort was undertaken to explore alternatives to the explicit wilful creation of geometry. Most notable in this field is the work of the late Paul Coates [Coates, 2010], and John Frazer [Frazer, 1995]. These proposed and tested radically different methods of design synthesis, often relying on novel low-level generation rules to enable a greater range of design possibles to be explored. Then, applying top-down methods such as evolutionary solvers to select effective candidates from the wide number of automatically generated forms. This approach could be compared to a phase in evolutionary development called the Cambrian Explosion. as [Valentine et al., 1999] summarises:

> "much genomic repatterning occurred during the Early Cambrian, involving both key control genes and regulators within their downstream cascades, as novel body plans evolved."

It was at this time that important features were developed such as eyes and the main body shapes emerged which was then improved on in later periods [Gould, 2000].

Importantly, this is a period of great differentiation and innovation. Later some of these early forms were selected out, in preference over a smaller number of the most effective forms. Similarly these computational methods stressed exploration and creation of new design spaces before optimisation.

One example method developed by computational design researchers, propose using units of space which act autonomously based on local interaction rules set up by the user [Coates et al., 1996]. These are akin to Turing's reaction diffusion methods [Turing, 1952]; whereby a system or organisation emerges from an initial unordered or undifferentiated state and the boundary conditions. Other similar methods [Coates et al., 1999] apply the concepts of tree development towards defining building construction. Here, the 'growing' rules are encoded, then the environment affects the building's growth before becoming a mature design. Whilst, typically, these rules are relatively simple, it is the complex interaction between the separate elements and the defined site/environment which can lead to a sophisticated resultant.

Similar but more applied research was undertaken by Frazer [Frazer, 1995]; Applying genetic algorithms to optimise systems that generate effective housing plans. In one case this was applied to aid users of self-build kits devised by Walter Segal, but also for the 'Universal Constructor' where a physical model was linked to a computational representation of the state of the model. From there metrics could be calculated on the current state and optimisation could be applied to suggest better alternatives. This work had a unique standpoint of looking at a different mode of interaction between machine and designer as equal participants in the same task, each learning from one other. These ideas were explored to quite some depth by Cedric Price for his generator project, a mixed-use corporate retreat in Florida, where the intention was that the use of a site would be defined by the interaction with a planning computer and realised by reconfiguring a modular building system based on the computers designs [Price, 2003]. A system was proposed by Price and Frazer that the computational architect system could get 'bored' and propose new designs to engender greater interaction with the system [Steenson, 2014]. Although the design was not realised many of the systems were prototyped by Frazer.

This research has also been extended by John Gero [Gero and Kazakov, 1998] and others at the same lab [Jo and Gero, 1998]. They have developed work that presented methods of automatically generating forms such as room layouts and beam cross-sections. An important development presented in this work it is that not just the phenotype can be evolved in this manner, but also the transcription rules. The transcription rules being the logic or process that converts a genotype code into a phenotype. Meaning the evolution of the evolution of the designed object can be included in a search or optimisation routine. Creating an evolving system with a resemblance to the way that the genotype's DNA code's emergent transcription is optimised based on the outcome of the phenotype. This is an important step in the development of these methods, as it can be shown that they are not to be bounded by the initial rules set up by the designer or programmer [Bentley and Kumar, 1999].

Gero makes an important differentiation between *search* and *exploration* [Gero, 1994]. Here he makes the distinction:

> "search is a process for locating values of variables in a defined state space whilst exploration is a process for producing state spaces".

This refers to exploration or the modification of the range of possible designs as being one core property of the act of designing. Where as search or optimisation is a much simpler process being bounded already.

These types of approaches have been referred to by some as 'computational morphogenesis'. And despite popularisation in digital design literature such as by De Landa [DeLanda, 2002], who explores how these methods could in theory enable a more fluid approach to configuring a design in the same way that evolution modifies the body-plan of an animal, few of these methods have been meaningfully applied to real design problems and arguably have reduced in interest since the introduction of parametric tools.

Some notable exceptions are the work of Derix and Hanna, who have shown these methods used on master-planning [Derix, 2009] and building [Hanna, 2007] schemes respectively. Some of which exhibit sophisticated solutions to complex problems, which, would not be possible without such approaches. However in the main these more atomic approaches seem to have been ignored by the general computational design community.

It is the belief of the author that this is in part due to the 'dead-end' nature of their current implementation. Most of the implementations of the systems above are realised using sophisticated computer coding, and built as monolithic custom programs without interfaces or plug-ins to typical CAD production software so reuse is problematic.

The author's own master thesis in this field is one such example; This work looked at developing novel L-system descriptions of geometry using basic metrics for solar collection and structural efficiency linked to a genetic algorithm optimisation [Joyce, 2008]. The study explored what forms were produced with different weighting factors placed on solar efficiency, gravity loading or wind loading producing forms similar to existing plants. This was implemented as a single Java app, with minimal ability to interact with the design process without having to modify the source-code and recompile. This example would not lead itself to collaborative design or use by others unless they had a high level of familiarity with computer science.

This makes these systems hard to apply to real design problems as in practice, integration is an important factor in making such methods relevant to a design process. This must be contrasted to similar methodologies which have been effectively applied in fields such as computer engineering; Methods which directly modify low-level systems have proven to develop complex systems which realise high-level goals. Examples such as neural networks which are now routinely used to build classifier systems or as integral in handwriting capture software [Russell and Norvig, 1995]. Arguably the most successful of such methods is Genetic Programming developed by Koza [Koza, 1992], which has been effectively applied in commercial packages and artificial intelligence systems, including generating code for some of the Windows 95 operating system.

Genetic programming represents a system which evolves mathematical functions or computer code by breaking and composing them using a tree structure like generative grammar, akin to Noam Chomsky's sentence trees for the human language [Chomsky, 2002]. By following the grammatical rules, it greatly improves the percentage of properly formed functions, but allows any operators and algebraic or scalar values to be used in the synthesising of the resultant function, expressive configurations that can be generated automatically. This approach has another great benefit that it produces output which can be understood. Whilst it may be complex, the output can be read, implemented or modified into other systems. This allows more conventional human checking and reasoning to be applied, which otherwise would be an alien approach, meaning such methods can be regarded as an automated code producing assistants, realising Paul Cotes dream [Coates, 2010]:

> "what if we could get the computer to write the code for us - so we can just write one program that evolves a program, and then we can all go back to sleep."

Applications in Practice

One applied example of machine learning is its use by the author on work place design. A part of Foster + Partners is a 'Work Place Consultancy' that focuses on providing recommendations and designs for the best use of office space. This service is often commissioned in-conjunction with a new building design commission, and works to configure the space (locating furniture, designation of room use, meeting rooms break out areas etc) so as to achieve the best utilisation, performance and enjoyment from the space and employees. It is interesting role that straddles aspects of spacial design and social engineering.

In one project an existing office for a technology company was to be to redesigned over a number of floors. Significant number of interviews and analysis were undertaken to capture the different needs of the various teams within the organisation. What resulted was a complex set of requirements and an over constrained design. This made the assignment of seating challenging to realise.

At this stage the author was involved in conceiving a system which could intelligently match teams to appropriate spaces within the floor plate. A methodology which clustered seats into groups was conceived using k-means algorithm. The implementation was based on purely spatial requirements creating effective seat groups. The algorithm could be tuned by selecting the number of clusters required, and typically due to the complexity of the problem the algorithm can return different cluster arrangements.

These clusters were then used to calculate metrics about the proposed space. This included basic properties like cluster number but also more social metrics; Such as the 'centrality' of a cluster, which is the sum of each members centrality, which is itself a spatial measure of that positions distance to every other position, represented a grid on the floor plan, and normalised with respect to floor area. Another metric was average intra and extra connectivity, which is a measure of the groups introvert or extrovert configuration, by measuring how many connections the individual cluster members have to those in the same or different clusters respectively, where connections are determined by a fixed maximum social distance.

By knowing the existing mix of groups in the company as well as identifying

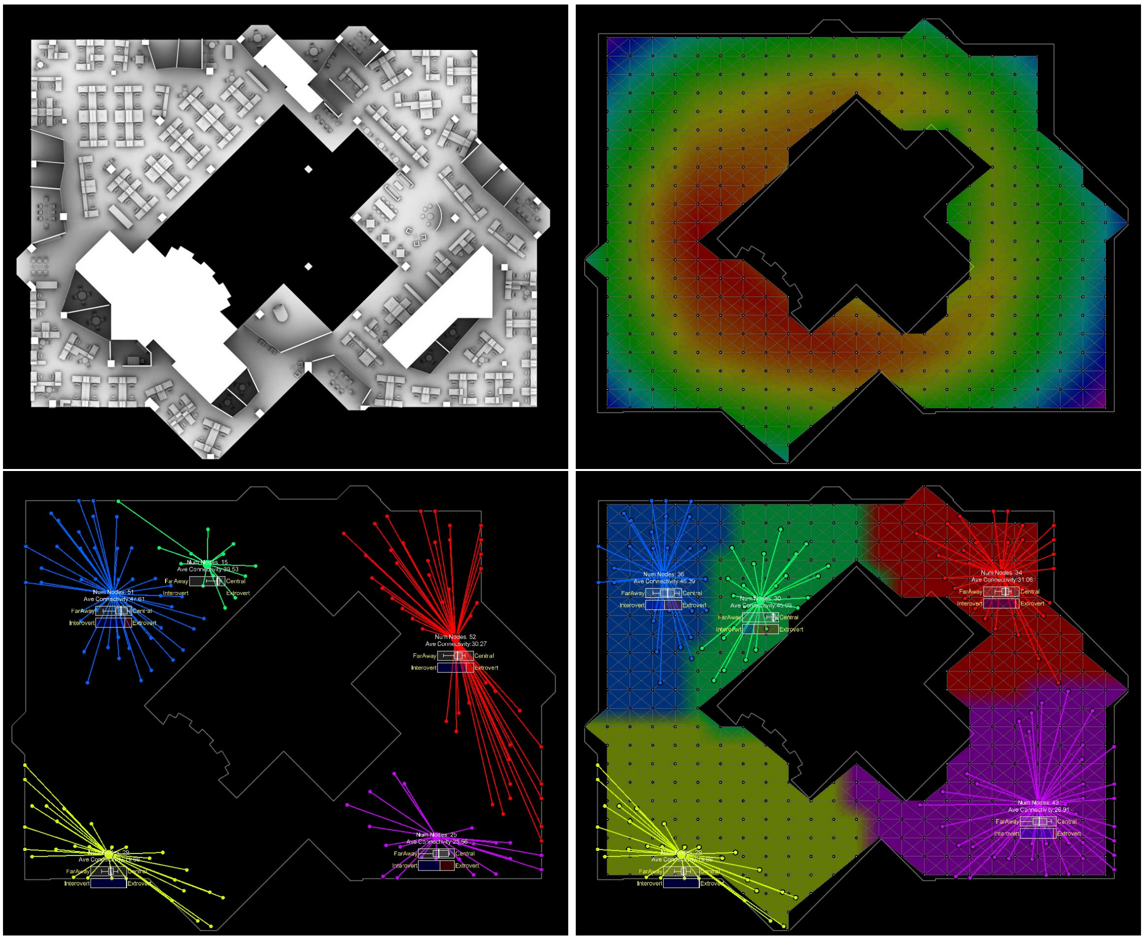


Figure 4-12: Example of workplace consultancy tool in action. From top right: Original layout, centrality metric for whole floor area, clustering with social metrics, automatically generated team zones. Source Foster + Partners.

needs of each group. E.g. introvert for I.T. department, extrovert for project management etc., a correlation between the configuration of seats as proposed and the resultant office utilisation could be derived. This minimised the effort of determining such properties manually, with metrics allowing for identification of adequate fit.

The initial interface was displayed on screen with an update button to apply and calculate the clusters. User eedback identified frustration with a lack of control to influence the systems results for external requirements. As the solver acts in a step wise fashion, it was possible to implement the ability to manually tweak and outright move cluster centroids, giving back control to designers, even allowing human-machine co-design.

Another issue was the lack of design development speed and usefulness owing to a basic but technically focussed interface. A desire was to engage in this kind of activity live with clients using it as a design tool but also a topic for discussion to gain more insight into their needs. A version of this was implemented, which rather than using a screen and mouse interface, worked using a camera to capture the proposed configuration. The system was able to identify floor plates by simple black and white Nolli plan, with each seat represented by round markers. The plan could then be redrawn or modified, as well as the markers removed added or moved. An updated version of this configuration with the metrics would then be produced and shown on screen.

Whilst there was considerable interest in the system, ultimately, it was deemed too technical and unpredictable to use in design sessions, and thus, its development was stopped. It was agreed that it worked reasonably well and could potentially save time in generation of good seating arrangements, and by having the metrics automatically, better performing solutions were identified more easily. However the lack of integration into the existing processes along with difficulty with its flexibility to respond to new requirements diminished its applicability. Fi-

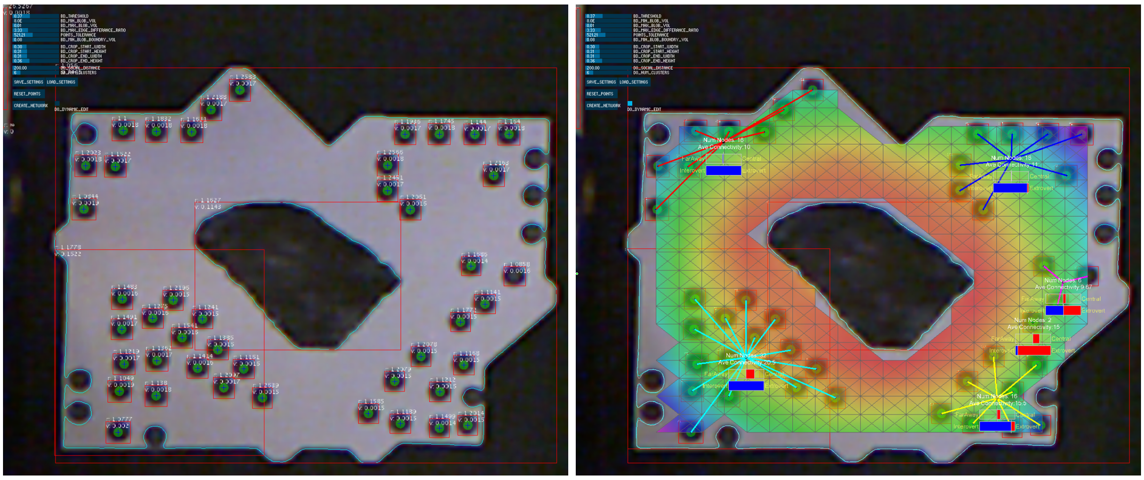


Figure 4-13: Example of workplace camera capture showing object recognition (left) and aumented analysis (right). Source Foster + Partners.

nally the visual quality was also an issue, as more polished user interfaces were required for confident interaction, especially if it was to be used with clients in a design workshop.

Meta-Parametric Design

Of interest to the author was demonstrating practical methods to improve the range of design search. It was felt that the principal systemic design approach (parametric associative systems) of enabling computational design and search was also the very thing preventing development into alternate options. Their rigid user defined parametric model was not very amenable to ease computational manipulation. However, it was also felt that these systems had many advantages as they were familiar with many designers of all levels of experience, and were actively used in the production of real projects and had many pre-existing plug-ins required for practical design projects.

It was identified that the core compositional feature of a parametric model was the associative graph. This represented the 'data-flow' programming paradigm, where information from one source or process was piped into another and so on.

These systems are able to identify when changes are made to input values or geometry, then via the dependency graph update any out-of-date processes. This dependency graph has a semi-lattice arrangement, as it does not allow for outputs of child processes to be used as inputs of parent processes as this would cause a cyclical dependency, and such arrangements can be called directed-acyclic-graphs.

The algorithmic generation of these graphs can be realised using a special type of genetic programming called 'Cartesian Genetic Programming' as was proposed by Miller [Miller and Thomson, 2000]. There is already significant research into their development as this is the same system arrangement that logic-gate design requires.

A study was undertaken by the author and John Harding [Harding et al., 2013], which investigated the potential use of such systems on the parametric modelling software Grasshopper, as well as outlining the theoretical and technical aspects of evolving parametric models. A proof of concept was developed called Embryo. It was a Grasshopper plug-in which allowed for a set of components to be selected then the plug-in could find a working combinations of these components.

Although an early study, it was shown that all or parts of a design logic could be replaced by an evolved system. By setting the input and output types to the parametric model and defining what desirable outputs should come of given inputs, there was a modular level of scope or control that could be given to the algorithm.

The configuration of the parametric model could be uniquely defined by a genome. This genome was independent of the process it was working on, essentially defining the operators and linkages as basic identifier numbers. The plug-in's ability was to ensure that these worked in Grasshopper, as unlike logic-gates, the input type (number, point, line, etc.) of the components was important in respect to that component returning its own valid output. An example of a bad definition would be passing a number and a curve as the two input points for a line definition, the output would not make any sense.

Figure 4-14: Samples of tower design options generated by Bjarke Ingels Group. Source [Harding et al., 2013].

Applications in Practice

Whilst a meta-parametric approach offers a much broader means to search the design space, these have to be accepted by the design team. In the reality of practice they have to be actively encouraged so as to allow for the active development of such systems which would be required to advance them to be practical. In Foster + Partners, this was not realised. Despite such methods presenting the potential of a much wider search of the design space, the introduction of section of unknown logic or 'Black-Boxes' to the process was not well received. The Bangalore project exemplifies a design process that allowed for significant variation in form and acceptance of high levels of computational control of the process. However, this is not a typical situation in the office, typically the design process is highly controlled and options generated are done so in a wilful way. Whilst this more radical generative approach was proposed by the author and the ARD team for various projects, these were met with understandable resistance when contrasted against a predictable controllable outcome, especially by conservative project managers. However this is not the case in all practices.

Other collaborating researchers have been able to apply this approach to the development of a number of early stage volumetric studies working in collaboration with Bjarke Ingels Group. In these cases this meta parametric approach matches up well with BIG's more exploratory unstructured approach to form-making. Indeed both the level of detail possible and the methodology is like a digital equivalent to BIG's method. Their design process involves creating numerous simple low-detail concept models out of foam, then in design reviews choosing favourite models and then looking at how the aspects of those can be combined or changed. This process then repeats until the group are happy with the result. This definitely aligns well with the evolved forms that are possible using the Embryo paired with a genetic algorithm.

As a result, Harding has taken this study to quite some depth in his EngD thesis [Harding, 2015], with the plug-in being further honed over the cycles of use in practice. Whilst the author has been able to use less of these techniques, it is believed that they are important step in computational morphogenesis. This is because they represent an approach which is more readily understood by the user, applied and integrated into existing process and work-flows, and can make use of the wide variety of plug-ins and tools already developed for parametric tools. But there is still some 'soft' issues to resolve if their adoption is to become mainstream.

4.2 Understanding

With the advent of computational design and stochastic solvers, more and more data related to design has been generated. Whilst manual techniques both in drafting and analysis placed practical limits on what could be calculated or drawn, it is now possible if not typical to generate more data than it is possible to humanly process over the period of a design. Data can be both intensive and extensive; intensive with lots of information relating to one specific aspect of design, a clear example being the data related to a FEA, with deflections and stresses being generated for each and every node and element. On the other hand extensive data is abound, for example in BIM models, a plethora of relevant and irreverent data can be acquired, from the overall length of all structural elements, gross floor area to the number of toilets. Much of this data can be further compounded together, all of which may or may not be vital, for example the cost per square metre (useful) vs number of toilets per column (not useful).

It is the experience of the author that the metrics chosen and focused on strongly affect the design process and outcome. This is natural as only by filtering can effective decisions be made. However, there is also the danger of underestimating or ignoring other issues, with dire consequences later on if they are critical to the safe performance of the building.

Edward Tufte has long been aware of the subtle power of data both intentional and accidental. He has highlighted the pitfalls of using data visualisation [Tufte, 1997] as well as championing approaches to making data clearer [Tufte and Graves-Morris His methods prioritise clear concise visualisations that have narrative and purpose. This is effective in the case of newspaper reports, and scientific papers, as both have a singular message for the data to augment and improve. In some aspects of design, this is also the case. A compelling story, which is supported by the necessary data and analysis is important in getting team members and client to buy-in to a specific proposal. Especially, in the cases where what is required is a move away from an existing solution.

Since Tufte's major works, much emphasis has been placed on data visualisation by the wider graphics community. With the advent of computation and data-bases more and more information is available to be understood. This has increased exponentially with the internet and even further the internet-of-things [Zhang et al., 2008]. New infrastructure and visualisation has been developed to support this scale of data analysis such as code framework Pandas for Python or

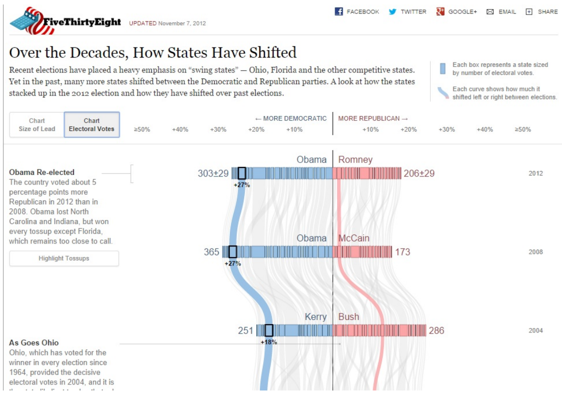


Figure 4-15: Example of New York Times on-line article using D3 data-driven-documents. Source New York Times

IBM's Watson. in addition new efforts to help people understand data have become popular, like the work 'Information is Beautiful' [McCandless, 2012], which has created a new proficiency in visualisation both creation and consumption.

The medium used to consume data has also changed predominantly from paper to on screen. With this new possibilities for expression including data understanding have been uncovered. One advancement consistent with Tufte's goals is the web visualisation library 'd3.js', developed by Mike Bostock a member of the New York Times on-line team. Data driven documents or 'D3' binds data to the web objects that represent them, almost like a BIM system for data and visualisations. It enables visualisations to transitioned into others at the click of a button, even dynamically modified to add, remove or updated. This has been applied to news stories, allowing easy access by lay users to easily manipulate large complex data sets through the web browser. This has gained much attention and interest with visualisations of elections, and the library is open source and used by many others.

What will be shown next are examples of the use of visualisation by the author in practice.

Figure 4-16: Overview of the Thames Hub proposal, including airport complex and integrated transport links. Source Foster + Partners.

4.2.1 Data Visualisation in Practice

One example of almost pure applied data visualisation was undertaken by the author for the Thames Hub project. Thames Hub was a proposal for the Davis Commission from a consortium of companies including Halcrow, Volterra and lead by Foster + Partners. The Davis Commission was an independent investigation into the future development of aviation for London, which is reported to the government. The commission was initiated due to the high utilisation of London's largest airport Heathrow (over 98%), with pressure from the main aircraft carriers such as Willy Walsh from British Airways to increase capacity. The Davis commission was the only major review of its kind for the capital and its recommendations were widely regraded to have a major influence on national aviation strategy and policy making in the next few years.

Before the interim report was due there was a public call for proposals to be

Figure 4-17: Plan of Thames Hub proposal on the Isle of Grain site. Source Foster + Partners.

considered by the commission. There were proposals from the main airports such as Heathrow, Gatwick and Stansted to extend their capacity with new runways and terminals, Foster + Partners also submitted scheme for a new 'mega-airport' on the bank of the Thames Estuary.

The Foster + Partners' Thames Hub project proposed a brand new site for air travel; based on extending by reclamation an area of current industrial activity on the Isle Of Grain on the mouth of the river Thames; 35 miles east of the centre of London. The proposal called for a large four runway configuration with room to upgrade to six, with high-speed links to London and potentially abroad via Euro-Tunnel, and an 'aviation-city' which would provide ancillary services for the air industry including sea-freight links The proposal also included wind and hydro power to be utilised via a new wind farm and tidal barrier in the river, with the barrier acting as a new river crossing to link north and east England.

It was the belief of the consortium that London need a new start in aviation; one that could be world class and in correct proportion to scale and importance of London on the global stage. This proposal was bold, and much larger in scope and vision than the favourite proposal of extending Heathrow's existing two runways

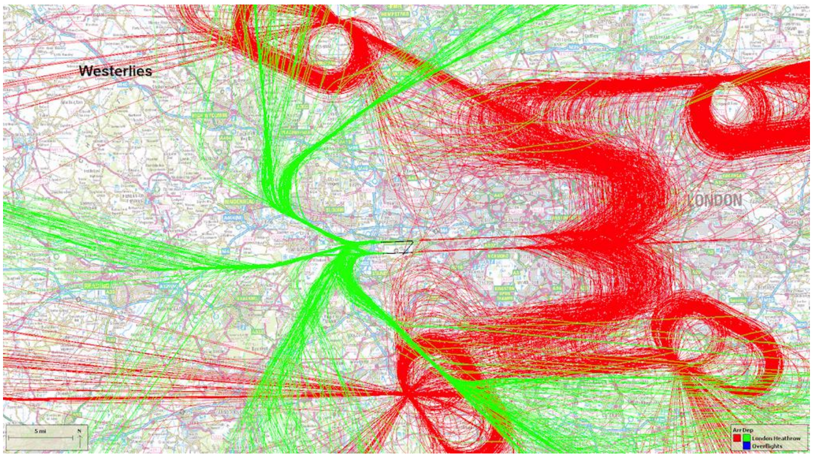


Figure 4-18: Flight paths coming into Heathrow from only westerly runway usage, green are take-offs and red landings, the loops are the holding stacks, flying over highly populated parts of London. Data from [Heathrow, 2014].

to three. It was felt that the Heathrow site was too tight and densely populated to extend without detriment to the local area. Especially, when including the impact to the large areas of urban London that were directly beneath the take-off landing and the holding stacks employed to control the volume of landing aircraft. Moreover this state would only worsen if one was to look at the rising demand both nationally and globally.

The grand scale of the project came from a position that the problem demanded a solution that would be significant and lasting, and that much of the air-traffic coming into London could be more effectively channelled into one 'hub', rather than sporadically into four airports. Over the period of concept development, there were a few key arguments that the Thames Hub vision relied on.

- Global demand for flights was growing.
- That growth wound not be diminished by the introduction (or not) of Thames Hub.
- Many would choose to fly via London over other destinations due to:

  A wealth of tourism options

A range of transfer detestations

However, equally there was resistance against such a proposal arguing that other existing mega-hubs already provided this role and that no improvement could be gained from moving more traffic through London.

It was felt that greater understanding of the implications of global mega-hub location was needed. This was undertaken with consortium partner Hambalt, a econometric consultancy who's specialism is in supporting logistic decisions for multi-national corporations. The studies aim was to understand how the attractiveness and potential use of London as a hub differed from other candidates such as Dubai, Istanbul, Beijing in a global context. This was to investigate if London could or should continue to be a central player in aviation given the economy changes over the next 10+ years, and also, the reasoning that London should defer its status to other emerging hubs such as Dubai.

An effective hub airport is one that has a 'hub and spoke' behaviour. It has major and frequent connections to many long-haul destinations (the spokes). As well as many connections to local destinations (the hub). Local destinations can be linked both by short haul connecting flights, but also other infrastructure such as road, train, sea. As such a hub links the short and long flights, so that people using it have much more choice of destinations at more favourable times. For a hub to work the important factor is the frequency of the flights, the more frequent a carrier has scheduled flights to destinations the more likely that a passenger can connect flight together with a reasonable layover 1-12 hours depending on destination. What results is a feedback loop were more passengers results in the carriers economically providing higher number of flights and to more places, which in return attracts more passengers. It was this suitability for hub status that was to be investigated, as it was the main premise of the Thames Hub project that London could support this.

Hambalt utilised their economic data, past and present, to build a range of key

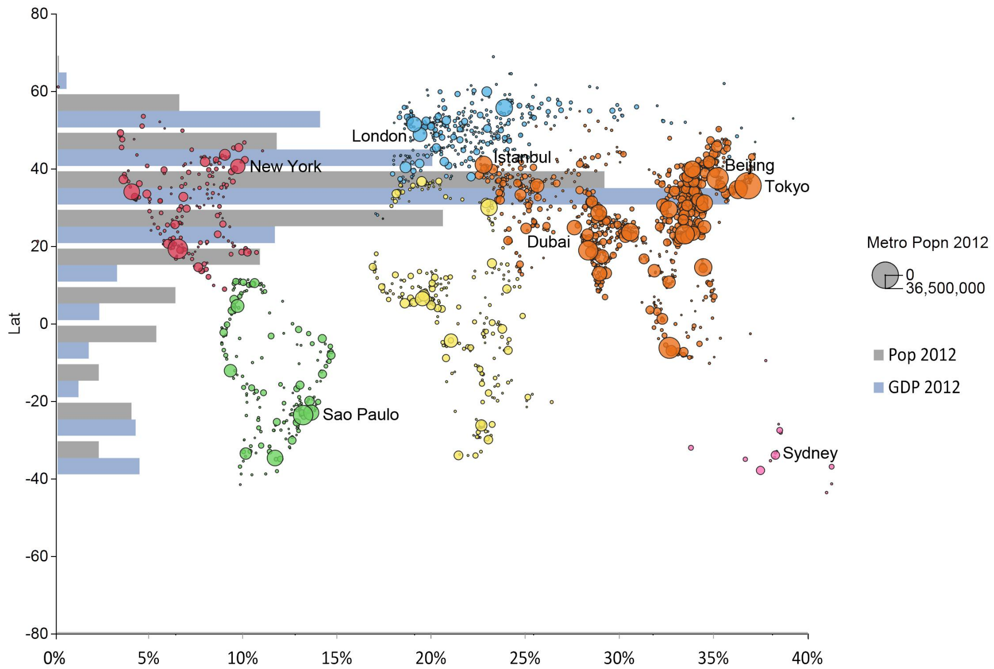


Figure 4-19: Histogram showing population and GDP distribution percentage binned by latitude for largest 2,000 cities. Cities shown on scatter plot with point area equivalent to the population in 2012.

indicators. These were also used to predict including flight demand over the next twenty years. With the data compiled at a per metropolitan area resolution.

However, this data set comprising of over 4600 cities represented a difficult prospect to meaningfully understand. Excel was initially used but there was a desire to improve on static graphs. In terms of interactivity, visual quality and ease of use, both for our own investigation but potentially for any public data.

It was decided to trial D3 to visualise the data. The development of these visualisations emerged whilst investigating the data with Hambult. By using web technology we were able to modify and introduce extra data and transitions during discussions and meetings. With the most useful and enlightening descriptions captured for use in larger reviews and public explanations.

A story emerged by exploring the data; by placing the world's GDP or Population or Fight Demand on one axis against the latitude, the bias towards the north

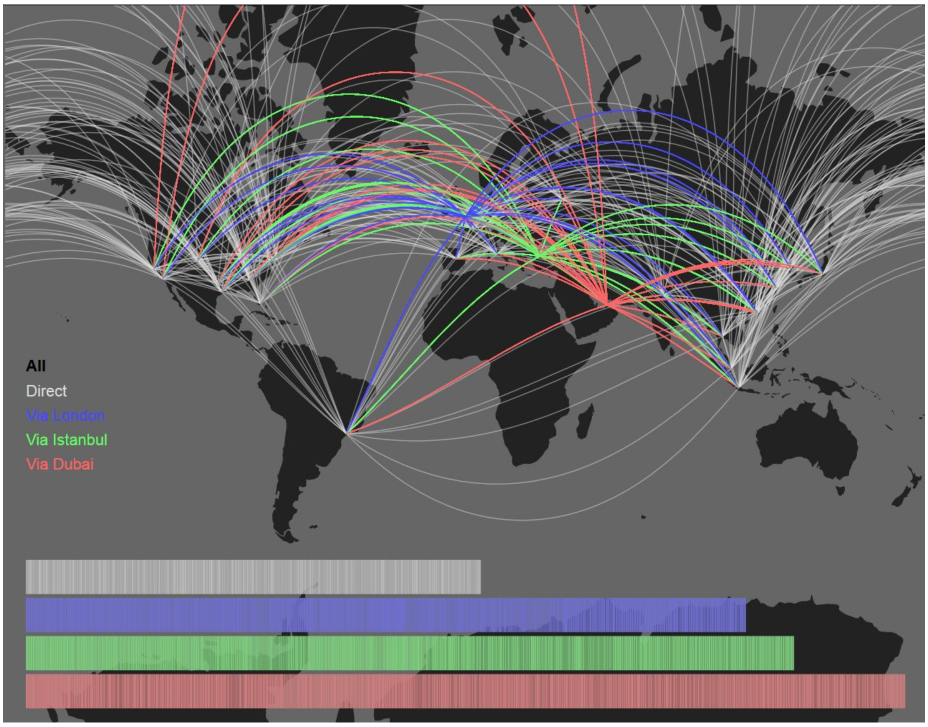


Figure 4-20: Visualisation of flight paths from twenty key cities direct and via London, Istanbul and Dubai in white, green and red respectively with a histogram of all routes by distance below.

was obvious figure 4-19. It was seen that an important role for hub airports was to connect this population, and that a aviation hub would be better off in the northern hemisphere, rather than on the equator, as is the case with Duabi. Thus, a connection distance study was undertaken, the result of which four main visualisations were generated.

The first visualisation Figure 4-20, is a projection of the great-circle or shortest path routes from the top 10 largest cities, as well as the top 10 major growth cities, to each other, as well as these routes via three main hubs. This was shown using the popular Mercator projection but also in an Orthographic projection, centred both from the north and south pole as shown in figure 4-21. This showed how northerly many of the ideal routes are, and how much longer and away from optimal, the routes from equatorial hubs had to be to support this location.

The second figure 4-22 reinforced this by highlighting the best hub in terms

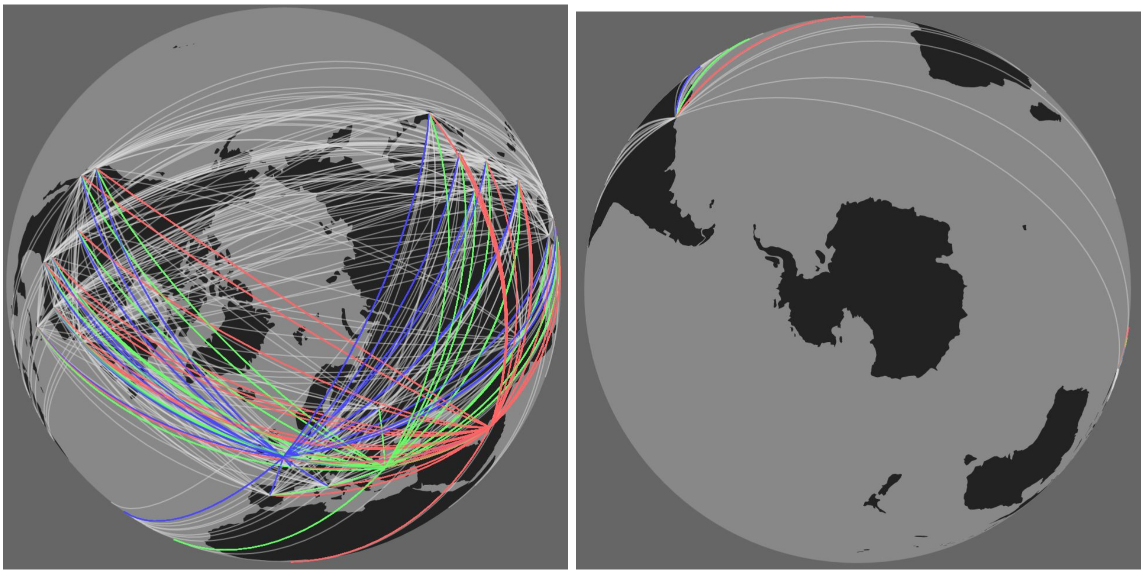


Figure 4-21: Visualisation of flight paths as in figure 4-20 but using orthographic projections centred in the north and south poles.

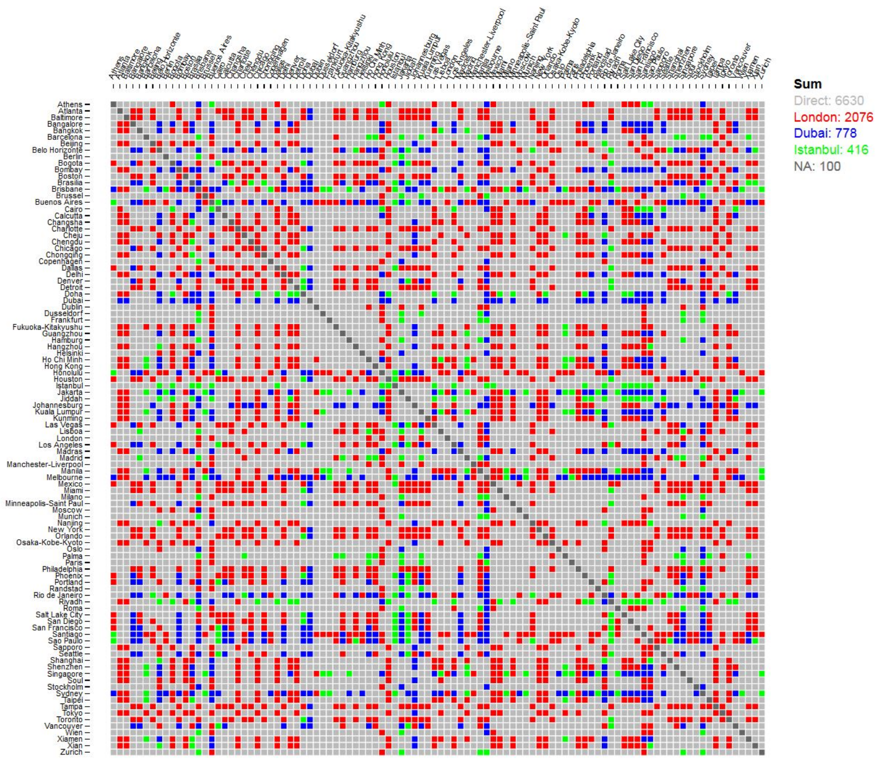


Figure 4-22: Matrix showing hub preference based on distance minimisation for the 100 largest cities, with grey squares representing city where a direct flight is less than 10,000km.

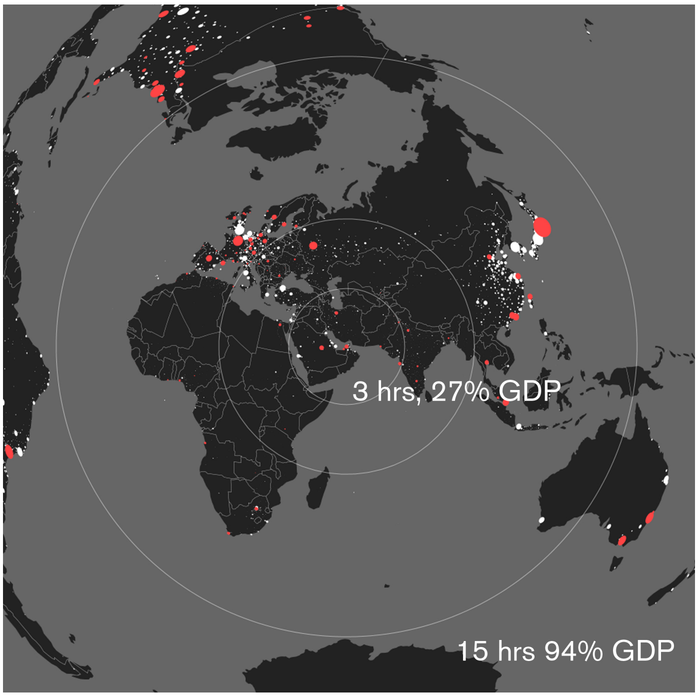


Figure 4-23: Azimuthal equidistant projection centred on London showing global metropolitan centres sized by GDP projected in 2020, with catchment rings of 3hrs and 15hrs and locations already flown directly from Heathrow in red.

of the great circle flight path distance minimisation via the three hubs shown as an origin destination matrix . Colouring each cell, representing a departure and arrival pair, by the best route in the the same scheme as the previous. This enabled a high level mass appraisal of route preference by visual integration of the volume of one hub/colour compared to the total. The d3 visualisation allowed the city rows and columns that where sortable by city size, continent or name, by the user so as it could be best explored and understood.

The third figure 4-23 showed the range of a short-haul and long-haul using a azimuthal equidistant projection. This projection has the unique property from of any potion of equal distance being an equal distance if both are measured starting from the centre point of the plot. Meaning that any radial ring represents the same distance from the centre, this is not possible on conventional 2D projections. In this projection, estimates for long and short-haul flight catchment areas where placed on the projection. This was overlaid with the top 4,600 cities based expected GDP

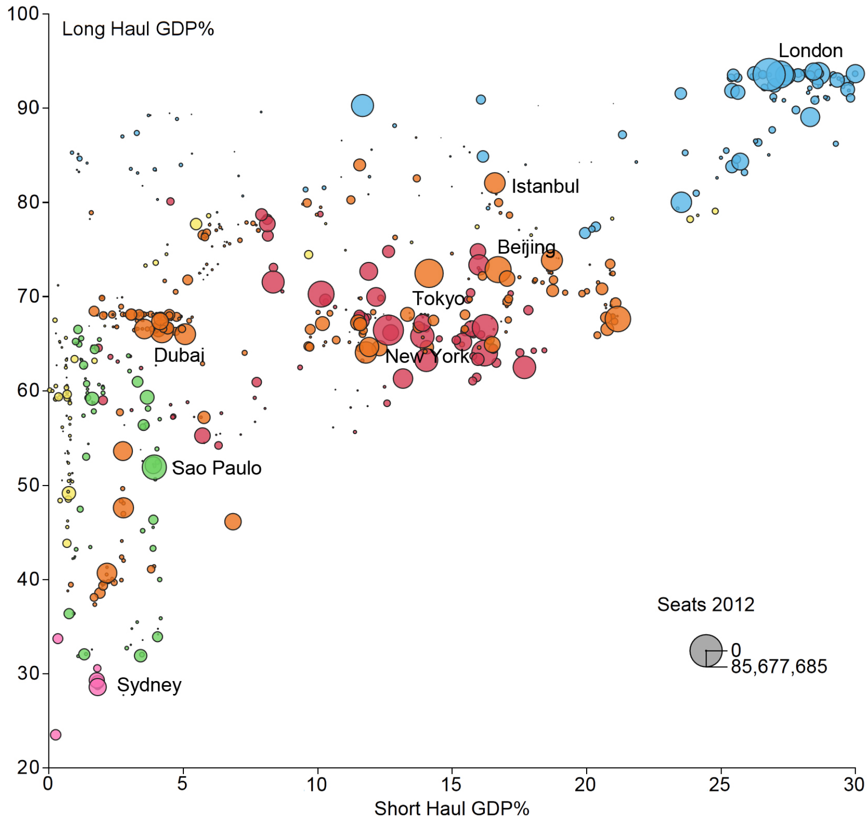
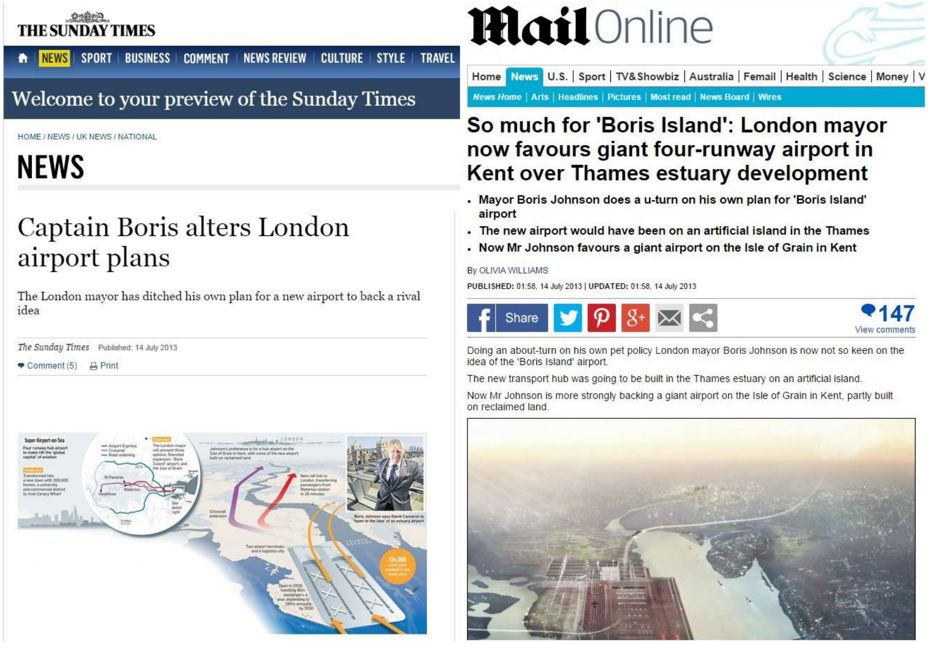


Figure 4-24: Global flight catchment data visualisation for long and short haul flights, data shows 1,000 of the over 4,600 data points (cities) used.

in 2020, to show the economic catchment area of an airport if completed before 2020. It was possible to select/click any city and have the projection update to centre this city. A special option was added if London was selected to colour the cities that currently have direct flights via Heathrow to show the 'white-space' cities that could be serviced/accessed if a major airport expansion was realised.

The forth figure 4-24 showed all the city data in a custom 4D multivariate visualisation. Starting with a conventional 2D scatter plot, but controlling the colour and dot size of each point to add more dimensions to be displayed. A mode was devised to enable one to click through plots of particular interest and add a narrative to the data. This provided a better understanding of the key data shown in previous plots but for every city, such as showing long-haul catchment against short-haul for each city. It also allowed users to generate their own plots from the


THE SUNDAY TIMES

NEWS | SPORT | BUSINESS | COMMENT | NEWS REVIEW | CULTURE | STYLE | TRAVEL

Welcome to your preview of the Sunday Times

HOME / NEWS / UK NEWS / NATIONAL

NEWS

Captain Boris alters London airport plans

The London mayor has ditched his own plan for a new airport to back a rival idea

The Sunday Times Published: 14 July 2013

Comment (5) Print

MailOnline

Home | News | U.S. | Sport | TV&Showbiz | Australia | Femail | Health | Science | Money | V

News Home | Arts | Headlines | Pictures | Most read | News Board | Wires

So much for 'Boris Island': London mayor now favours giant four-runway airport in Kent over Thames estuary development

- Mayor Boris Johnson does a u-turn on his own plan for 'Boris Island' airport
- The new airport would have been on an artificial island in the Thames
- Now Mr Johnson favours a giant airport on the Isle of Grain in Kent

By OLIVIA WILLIAMS

PUBLISHED: 01:58, 14 July 2013 | UPDATED: 01:58, 14 July 2013

Share

147 View comments

Doing an about-turn on his own pet policy London mayor Boris Johnson is now not so keen on the idea of the 'Boris Island' airport.

The new transport hub was going to be built in the Thames estuary on an artificial island.

Now Mr Johnson is more strongly backing a giant airport on the Isle of Grain in Kent, partly built on reclaimed land.

Figure 4-25: Press cuttings showing London mayor Boris Johnson backing the Thames Hub scheme over his own [Williams, 2013] and [Times, 2013].

many data categories that each city has, for free exploration of the data. Finally by displaying latitude and/or longitude on the plot axis data could easily be made spatial also.

These plots were reproduced in the proposal reports 'Global Connectivity' section [Foster and Partners, 2013], outlining why London is a better and more sustainable place for a hub, both due to more economic growth predicted in the northern hemisphere, as well as minimising air-miles flown by avoiding the sparsely populated equator. This means that London is naturally a good place to grow and support an infrastructure such as Thames Hub. Whilst these static captures of the data were useful, perhaps the biggest impact was its role in providing dynamic presentations to important parties. One key example being to present the findings to the Mayor of London Boris Johnston. This presentation and the argument for London as the best hub candidate helped persuade him to publicly endorse our scheme and argue for the initiative, using our findings on the bigger picture.

4.2.2 Methods to Visualise Performance

The Thames Hub project acted as an instigator to question how data was visualised more generally in F+P. The visualisations and key technology were re-appropriated into a more generic tool that could graph any data set with little to no preparation needed. The data structures chosen were comma-separated-values and the more object centred JSON data. Robust data parsing was used to allow for the system to plot data of both numeric but also ordinal values, allowing for plots of values as was well as groups to infer relationships and trends. Furthermore, handling of 'jagged' data sets where some objects have keys and values for those keys and others do not. This allowed the visualisations to be quickly applied to any data after reformatting to .csv or JSON.

It was intended for these technologies to have an application on design space exploration and optimisation data sets. Rapidly they where used by the author to visualise engineering model performance. Due to the ability to set the value for any axis from the browser data could be explored much more quickly than generating plots in something like Excel. The transitions between diffrent plots where also easy to include, and by observing data explanations by the likes of Hans Rosling, one could see the benefit in animating these changes as they helped to observe trends and better orientate the data by tracing its movement [Rosling, 2006].

Important in this field is also the work of Stephen Few, who developed many concepts for 'data-dashborads', a single page overview of carefully chosen metrics, typically developed to enable executives to make well informed decisions, the format can also be applied else where [Few, 2006].

Data Visualisation in Engineering Practice

This technology was applied to the previously covered Tocumen modal analysis data set which was a project being undertaken at a similar time [Joyce, 2015]. The data set used had over 700 points, each representing a different potential design for the link bridge. With the original raw data and basic excel graphing methods, it was possible to quickly confine a selection to only viable solutions by setting some boundary values. It was equally as easy to define the optimal solution for a given criteria or show the Pareto front for performance function comparison. However, the modal stability was harder to visualise having three main modes as well as other important criteria, this also made the data hard to understand. Moreover being able to correlate changes in section with modal behaviour was time consuming and involved navigating many plots.

By using the dynamic interactive viewer developed it was easier for the engineers to switch comparisons between different modes and appreciate the data more fully. This helped explore and understand the behaviour of performance functions to input values. Over the course of its use it was especially effective in looking at a plot of two performance objectives on the X and Y axis, observing the Paretro optimal points, and comparing this with input properties which could be described by both colour and dot size.

To provide more insight into the data, more features were added based on feedback from the engineers; This included the ability to select or hover over points to show all the information of that point. By highlighting or hiding individual or groups of data points, it helped in tracking the data or cutting down options when observing the data. It also helped in creating specific collections of plots that were found to be most insightful and could be circulated. In this way, well reasoned decisions could be made relatively quickly with a depth of understanding not possible before.

Overall there was a good response to this when used for projects. The major benefit was seen as producing more engaging presentations of engineering data which was often overlooked in preference of more visually compelling but arguably less important information. The ability to generate any graph also helped

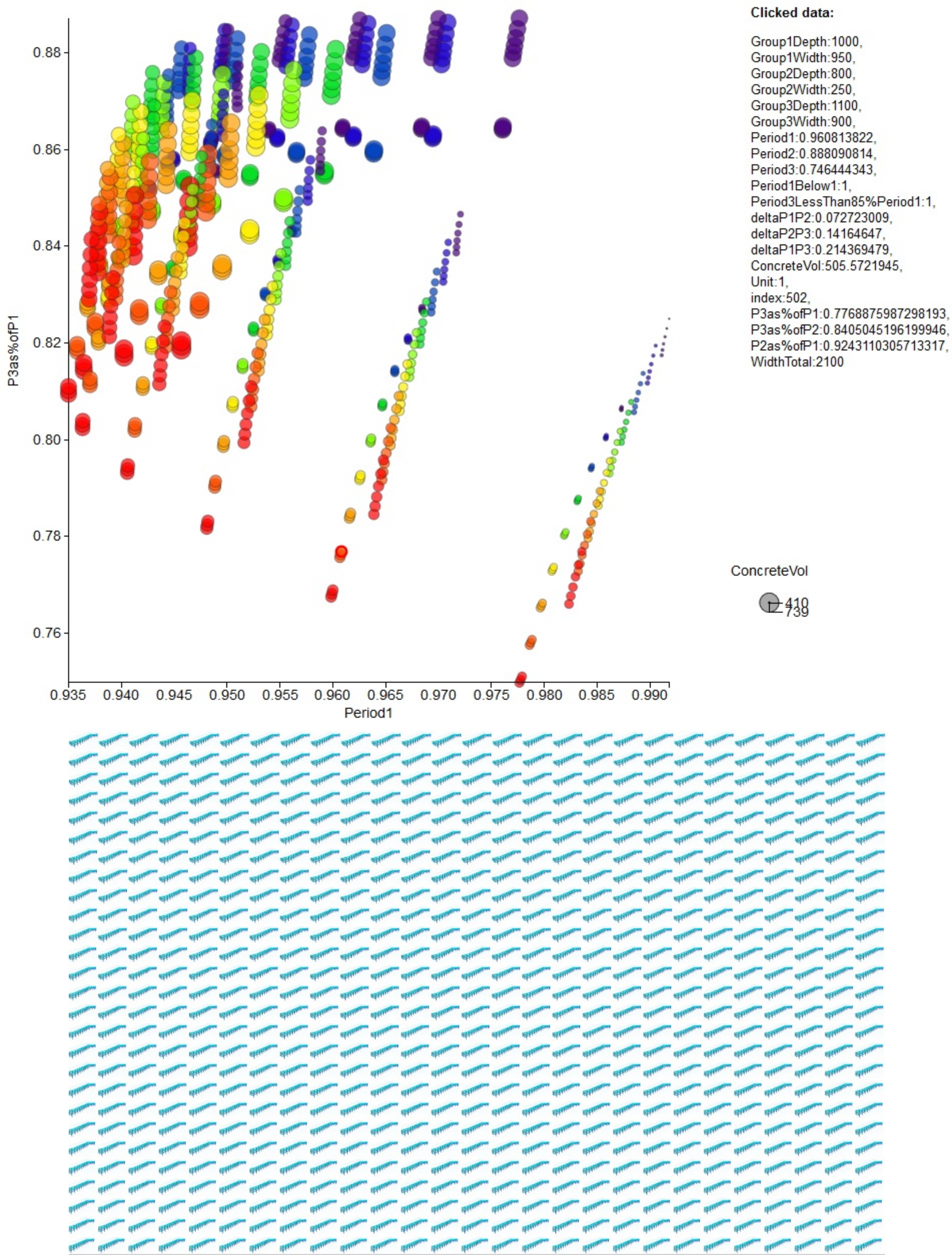


Figure 4-26: Visualisation of major trade off between first period and the difference between the first and the third period. Circle size represents concrete volume used, with the colour denoting the section size of one of three of the beam groups. Below is each of the designs plotted.

explore data quickly during meetings as questions came up. The web nature of data presentation allowed easier access via tablets, which are readily available in meetings. Furthermore the web pages were regarded as easier to engage with and 'drive' by senior design members, than excel sheets or static tables. It was felt by Engineers that it helped understanding of the data. However they also raised the danger of "playing with the data rather than understanding it" as well as "loosing the point of what the data was for".

There were some issues in deployment especially on iPads. Whilst the web pages were designed to run locally and were often presented this way on computers, this is not possible for mobile devices and they must be supported by a server. There was a simple web-server set up to enable access. Whilst the transfer of a page to this server is only marginally more complex than copying to a file, access restrictions and a lack of familiarity with using a web-servers prevented others from outside the ARD team making use of these without assistance. It would be possible to have a server that is set-up to receive this data and visualise it, but this full implementation was not called for, based on its current level of use.

If this approach was more widely used a level of security would also be required to remove the risk of others getting access to data. For example the office has now made the local network available via WiFi so they could be served from there. However this would not be a solution in some of our satellite offices, or for other other firms implementing this kind of technology.

Similar work has also been developed by others. For example the idea of a 'design-dashboard' by Rolvink [Rolvink et al., 2010], which attempts to bring together data pertaining to tall-tower design into one place. This is done with special curated visualisations which are closely coupled to the design of a tower, however, this is very specific and tailored to the problem and thus, requires a deep understanding of the issues and as such is not necessarily generic.

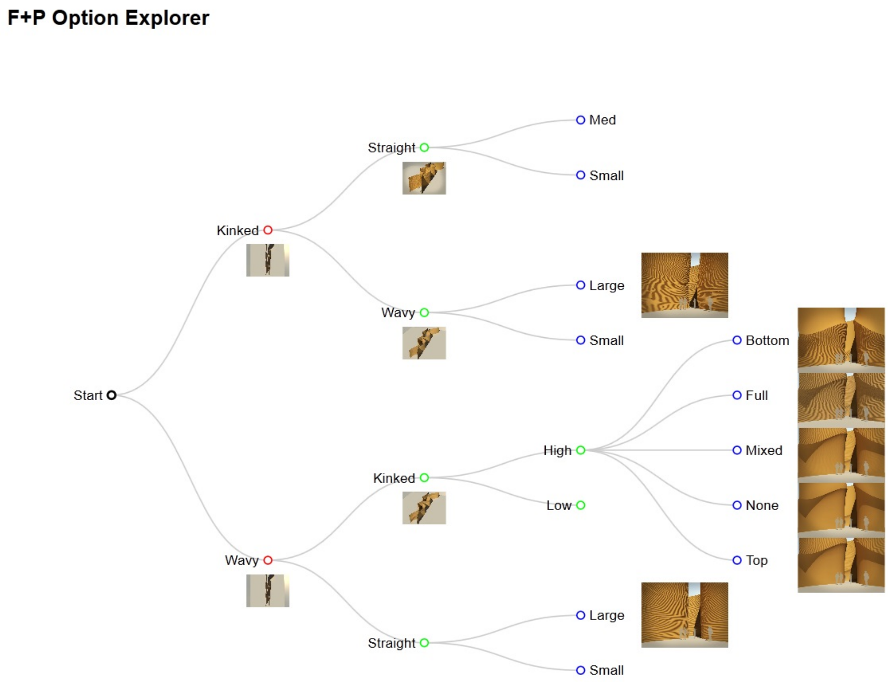


Figure 4-27: Example of the option-tree exploration web interface, with options expanded.

Visualisation for Option Capture

The creation of information display and dashboards for design is not new, however the application of the web for fast iterative development of interactive visualisation allows for quicker prototyping and trailing of visualisations.

One such study attempted to address the capture and progression options of an active design process. During the UAE Pavilion project many different options were considered; this was especially true for the Canyon section. This primarily related to the overall plan geometry of the walls, the sectional profile of the wall both type and projection distance, as well as the ripple pattern location on the wall and ripple size. The sheer number of these independent variations meant that the number of possible combinations could potentially be as many as 180 if no executive decision was made to rule out options. Given each version required manual modelling and rendering, there was value in understanding and minimising the

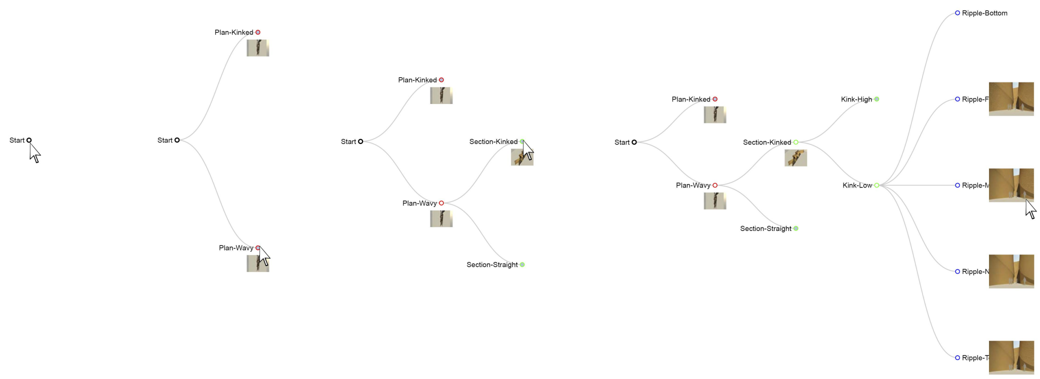


Figure 4-28: Example of the option-tree exploration web interface being used in a session.

number of options fully generated.

Owing to the subtle variations in the options and the desire for the Canyon section to be as evocative as possible the decision on closing down options was not taken lightly. Due to the different scales there was a natural basis for the order specific decisions to be taken in, as larger scale decisions had a greater impact on the overall aesthetic. However often some decisions required fully rendered images to make meaningful conclusions on the holistic visual effect. The difficulty was compounded by choices being revised and explained to senior designers and clients for approval.

During this process, an attempt was made to capture and record the options explored, to help decision making and hopefully reducing the work-load of those having to make the options. A visualisation that captures the impact as decisions are made was created. This is in the form of a tree, where each branch represents a differentiation.

The options were organised in a folder structure with each folder holding an image of the current state, and sub-folders representing further decisions after that. A custom Python script was then created which stored this structure as a JSON, which could be used by the web page. The web page was then created to allow a decision tree to be explored by clicking on the nodes to expand or hide sub options which then could be interacted with, to show more sub options and the progress of design development. Each node included an expandable image to view the current result of the series of decisions. This could then be easily updated as new options were added. A simple text file was included in each folder to allow for naming and extra information to be passed to the website if required.

This was used during the session to aid exploration of options already rendered. This helped internal record keeping by quickly reviewing what had been produced which prevented regression on decisions and time spent repeating work by reminding the design team why some options were preferred over others. It also allowed the team to understand how many options they considered and appreciate how many more they would be asking for.

This approach is something that shows how options could be better captured for qualitative decision making. The nature of design development is well aligned with storing options in this way; Presenting an engaging and more accessible way to revisit decisions.

4.3 Conclusions

This chapter has shown that it is possible to automate the search of a given design space. With the introduction of intent capture by parametric systems, or more generally by meta-processes, one is able to remove much of the processing and manual intervention required. This significantly widens the number of designs or data-points that it is practical to generate.

By introducing parallel or cluster computing tasks that were previously thought unreasonable to undertake on mass can also be tackled. Importantly, this can be done with relatively minimal impact on the original process. By reproducing the environment on multiple machines and managing the workload of each, the original Grasshopper model(s) or similar can be scaled up.

This kind of search can still be limiting, and attempts to assist the act of computational exploration of new aforementioned parametric models has also been demonstrated. However, the acceptance of such approaches into a collaborative design effort is still lacking.

Utilising these methods opens up new challenges in visualisation of such data. Examples have been shown of both internal and external presentations using web interfaces to convey understanding better understanding of the available data, by using the novel interactive mechanisms made available by the 'web' medium.

Chapter 5

Consolidated Proposal on Strategies for Performance

> "if you try and take a cat apart to see how it works, the first thing you have on your hands is a non-working cat."
>
> Douglas Adams, 2002

In previous chapters, we have discussed existing and new methodologies to support performance driven design processes. This chapter will show some generalisations that can be made with these case studies; Highlighting how these projects have been assisted (or not) by technology, as well as looking at the socio-technological interactions. It will aim to demonstrate how there is a common methodology that can be both generalised and specifically targets individual project requirements.

5.1 Observations and Analysis

Over the series of studies and experience at Foster+ Partners the author has identified some main themes and trends that have prevailed through these case studies.

- Integration of performance metrics
- Automation of design processes and data/analysis processing
- Higher-level control over design processes
- Design as a process of search and exploration

It is possible to place these themes within the 'Design Prototypes' schema proposed by Gero [Gero, 1990]. In the Design Prototypes schema, design is defined as the process of going from a set of functions, F, to a structure, S, finally producing design documentation, D. Functions (or requirements) are transformed into a structure, $F \longrightarrow S$, by a process of formulating expected behaviours, B_e, from the functions, synthesising these to a structure (where a structure is understood as a design instance, such as a building). Then, this synthesis is analysed to calculate/determine a set of actual behaviours, B_a, which can be compared against the expected behaviours $B_e \longleftrightarrow B_a$. This comparison leads to reformulation of both the structure, expected behaviours and the desired functions. In this way the structure leads to an indirect feedback to the function requirements via these three reformulations which represent the 'situated function-behaviour-structure framework' [Gero and Kannengiesser, 2004] and is shown in figure 5-1.

This schema is important as it emphasises the challenge of the design process, and the complexity of design conception over simple design representation. It shows that attempts to improve design process and understating, represent a larger challenge, as they require aid to analysis, comparison, synthesis and reformulation. This can be compared to BIM in its current form, which only helps with documentation part of this schema.

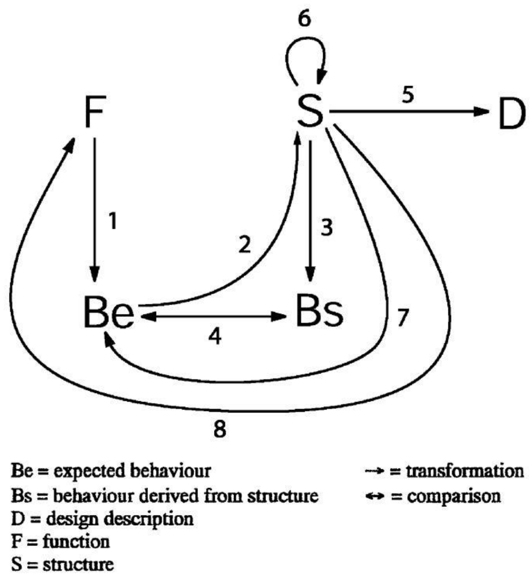


Figure 5-1: Situated Function-Behaviour-Structure framework transition diagram. Numbered transitions are 1:formulation, 2:synthesis, 3: analysis, 4:evaluation, 5:documentation, 6-8:Reformulation. After [Gero and Kannengiesser, 2004].

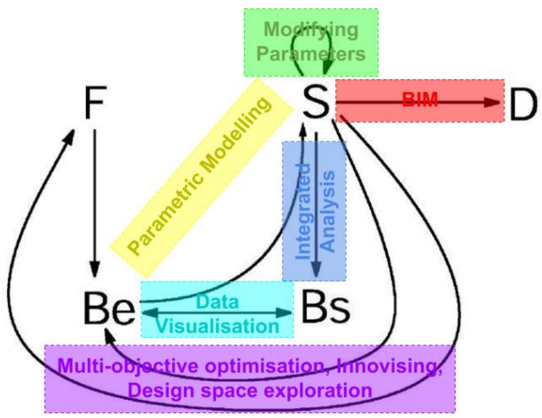


Figure 5-2: Function-Behaviour-Structure framework highlighting technologically supported translations. Adapted from [Gero and Kannengiesser, 2004].

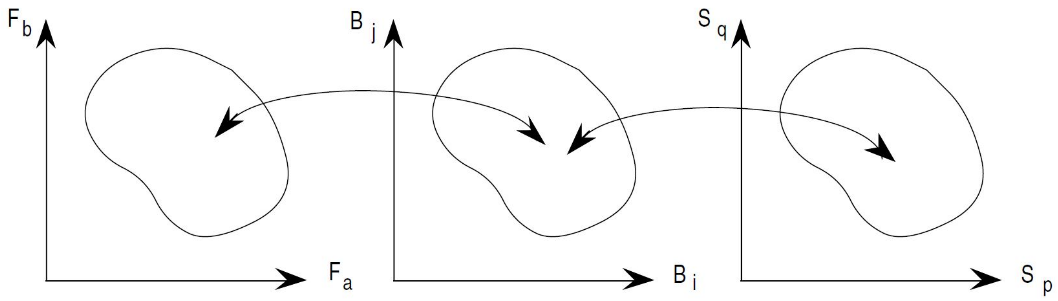


Figure 5-3: The three subspaces of Function F, Behaviour B and Structure S which constitute as state space of a design. After [Gero, 1994].

It is believed by the author that the aforementioned studies address methods of aiding the transitions defined in this schema. Parametric design assists in documentation and generation of models for analysis. It could also be argued that this can in some instances directly couple the F, S and B_a together, by introducing geometric rules which link requirements directly to observable outcome by geometric constructs. Integrated and automated generation and data extraction of analysis helps in the analysis transformation $S \longrightarrow B_a$, as well as evaluation $B_e \longleftrightarrow B_a$. Data visualisation aids in evaluation, as well as understanding required for reformulation. Automated design creation assists in the synthesis $B_e \longrightarrow S$, and design space exploration aids reformulation. As such an augmented version of the FBS framework, with these proposed technology assisted transitions, is captured in figure 5-2.

5.1.1 Design and Optimisation as Exploration and Search

It has been seen that design objectives change based on the design stage, and, the type of questions that require answering. It is the view of the author that design is often a process of exploration as Gero explains [Gero, 1994]. Here, Gero defines *search* as the process of finding a single specific value that satisfies the goals set, where as *exploration* is the process of finding out the constraints or limits to the design space, in part to find what is possible but also to define what is desired for

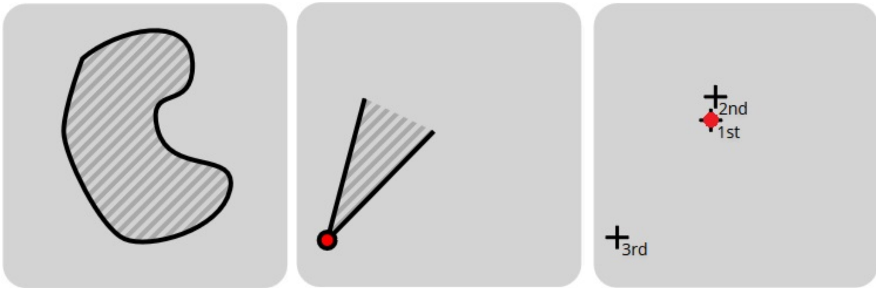


Figure 5-4: The stages of design support: boundary defining, directed exploration and decisive searching

the design.

It is possible to divide design activities between exploration and search stages. Typically early stage exploration will transition over time to search, as decisions are made and design options are fixed. This is done for pragmatic reasons as pure exploration would result in an unresolved design.

This exploration and subsequent fixity requires the capability to adapt to these needs. It is believed that the process of defining a design comprises of stages of expanding, exploring and progressive restricting of these three spaces FBS. These distinct behaviours are shown graphically in figure 5-4

It has been shown that optimisation approaches such as genetic algorithms can help to improve the later searching phase of a design/structure space. However, without a well defined function space, results can be disappointing as results often fail to fulfil basic requirements or aesthetic needs. These methods do not help this issue, as they offer little feedback to expected behaviour unless iterated themselves, becoming a more sophisticated version of the structure to actual behaviour transition. Multi-objective optimisation has shown to offer better feedback, with accurate reformulation realised for behaviour space.

Within the tightly confined synthesis of form-finding methods, effective directed exploration of actual behaviours can be realised quickly. These are reliant on a specific formulation as well as a strong understanding of the expected behaviour/outcome, as these methods dictate the end result and can be a waste of time if they do not deliver.

Finally the most well aligned with Gero's concept of design, is large scale design space exploration coupled with visualisation. This is where a design/structure space can be widely sampled to generate wide links to the behaviour space and performance metrics. At the same time by concentrating on the behaviours from a whole parametric model's range, and decisions to change the state space (parametric deign) or function space (requirements) can be made.

5.1.2 A Sliding Scale

It is believed that these ways of navigating and changing the design spaces represent a spectrum of design needs and has been shown these need a different palette of tools to support them. Tomas Mendez in his thesis [Mendez, 2014] points to a middle ground in his differentiation between *search* and *exploitation*:

> "Exploration is the process responsible for covering the entire search space, include the vast majority of solutions in the search.. ..Exploitation is the process responsible for signalling out the best performing solutions, to direct the search process towards promising areas and generally reduce the search space by focusing on the best solutions."

Mendez's definition of 'search' is more analogous to Gero's definition of 'exploration'. However, 'exploration' shows how optimisation may be used to both explore and narrow a design space simultaneously, by using a optimisation to find new but targeted options.

5.2 Progressive Performance System a Kit-of-Parts

The previous analysis of the design process of exploration and search shows that design rather than being a monolithic undertaking is actually a composition of many different processes, as well as decision forming and making activities. The tools developed have been created in such a way to integrate with digital design flows rather than dictating them, enabling such tools to act as a 'kit-of-parts', and reconfigurable based on the problem. The benefits of which are described by Aish [Aish et al., 2013]. It is of value to understand how these are actually used in practice to derive the requirements placed on them. To appreciate the addition that the previous work has contributed (or not), to this system of parts, a brief review and analysis will be undertaken of each section, emphasising its place and contribution to a computationally assisted exploration and search system, deriving the requirements for such a system which will then be applied in practice.

Modelling and Representation

The modelling and representation approach is of central importance in an effective system. This more often than not, defines the limits of what is tractable to create, control and consider. With the decision as to what representation software/tool to use having an influence on what is possible to create, and more specifically, in a practical context, what is effective and expedient to construct. With the tool influencing designer, and thus style, as has been shown by Schumacher [Schumacher, 2009].

To this end it has been shown that the more flexible and open a system is, the more degrees of freedom and links between software can be exploited. Currently, parametric modelling represents the most widely applied system in this respect, with Rhino's Grasshopper platform used in almost every case-study undertaken by the author. Predominantly, this has to do with the speed and ease of building representations in the system. The ease of interoperability of Rhino files and the body of plug-ins and extensions reinforce this usefulness.

For a modular system, no one platform should have a monopoly which could act as a bottleneck to development. It has been observed that systems used are changing; when the author was first at Foster + Partners over 6 years ago, most complex geometrical projects were realised in Generative Components or via the Microstation VBA api. However, during the last three years, the author has not had cause to use this more than five times and then mostly for supporting older projects reliant on it. Even now with Rhino being a dominant platform, Dynamo, the parametric modelling system for Revit has been gaining populatiry. It has be adopted in use for specific projects by the author and others at Foster + Partners, to support Revit and BIM projects which are coming into favour more and more.

The parametric paradigm has become almost ubiquitous for complex geometry creation. Whilst this standard has the benefits of ease to transfer skills owing to the similar paradigms used, there is also a danger that this limits the underlying range of design possibilities, as the author has discussed [Harding et al., 2013]. However, equally the author has shown that these systems can be expanded either by custom extension, or by collaborations with the developers, as the author has done with Design Script [Aish et al., 2012].

Integrated Analysis

The ability to convert from geometric representation to engineering analysis rapidly, is a key factor in integration. Effective methods have been shown to be robust and simple to implement, such as with the Madrid Stadium case-study. The same case-study highlights the flexibility required to adapt integration, to introduce new complexity such as pre-tensioned ties or modal analysis results. Whilst the author has implemented bespoke targeted interfaces, it was with the development of the Foster + Partners hub that a level of generality and integration into existing work-

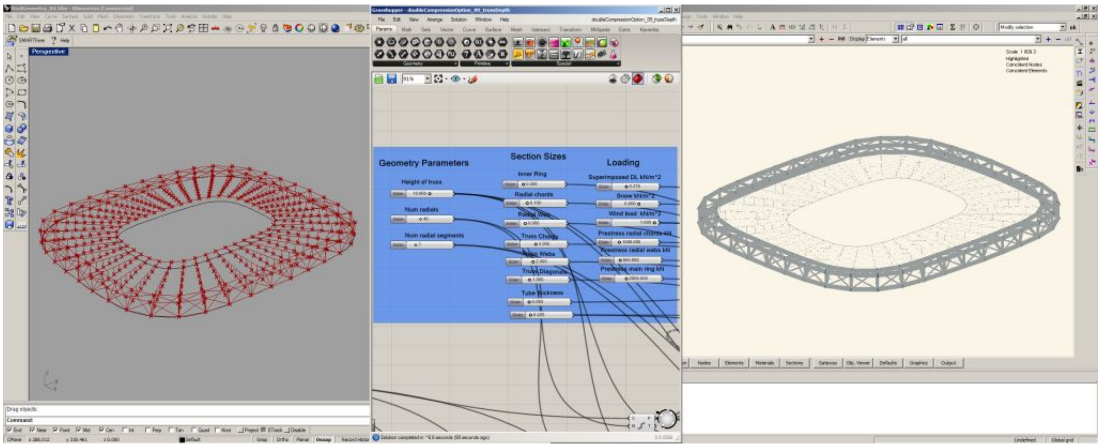


Figure 5-5: Example parametric work flow for a stadium ring design creating geometry and structural models

flows that the effectiveness of such tools can be found.

These transfer systems must be capable to support different engineering software used, as it has been found that engineers prefer and trust some packages over others. To this end the idea of having a central system is useful to reduce the potentially large effort of programming 'hub-to-spoke' links rather than a system to system link.

The author has found with such a large development task, as well as the upkeep of the various links as software is inevitably upgraded, this workload can be significant and is in quite a different nature to project work although development requirements will often come at the same time. It is believed that this kind of interoperability tool may be best implemented by external parties; as with Geometry Gym [Mirtschin, 2011] or potentially by a consortium or open-source effort of interested parties. Subsequently the author has shifted to using Geometry Gym for such work-flows.

Implementation details aside, the most important factors in such systems are the ease and flexibility to interact with different platforms and the ability to extract data from the analysis quickly. These features allow it to be used within the context of rapidly changing project demands.

Automation

The UAE panel geometry creation process showed how computational processes can be scaled up effectively to run complex procedures in parallel with relatively light weight management. The Tocumen project went further to show how automation can be used to save time and effort when undertaking the exploration of the design space. This required the ability to integrate analysis and extract high-level information from each instance of the models.

Whilst it is possible to use parametric systems replication/combinatorial features carefully to produce multiple models this is far from ideal; The complexity of building such complex multiple systems without accident is difficult. Moreover the current parametric systems currently solve iteratively in series making them slow. Whilst these are very much parellisable at theoretical level, with plans to do this in Grasshopper [Rutten, 2014], this could potentially causes problems for interacting with engineering software instances.

Instead it has been found that writing meta-processes that orchestrate systems from a higher level is more natural to understand, in keeping with the single user design session that many of the dependent systems are designed for. The ability to run these systems multiple times on the same machine, or on multiple machines or both provides a mechanism to scale up processes without having to resort to cloud solutions.

This does however require simple expressive ways of manipulating key systems via script or macro. In this way the current systems have much to learn from the Unix approach of combining processes into larger scripts, as the author has mentioned in [Tsigkari et al., 2013].

Optimisation

Whilst the work of Gero has shown that optimisation is not as central in design, as it is in engineering, it is argued that for conclusive periods of the design process they can be very useful. Their application during the UAE panel rationalisation is indicative of this.

Also the use of optimisation does not have to run contrary to design exploration. Optimisation at the early stage, to find higher performing alternatives is an effective strategy as discussed by [Deb and Srinivasan, 2006] and termed 'Innovisation'. This is the activity that was undertaken for the Bangalore project to find new forms, and has been observed by [Bradner et al., 2014] in other architectural practices.

Projects such as the Mexico Airport roof, has shown the application of specialised structural rationalisation methods to be effective in design processes. These are less general-purpose and more for specific applications or functional requirements. Whilst they are not applied on most projects, it is important to have them integrable into a system, as they can effectively short-cut to good solutions, if the resultant aesthetic is desired and lack of control is accepted. In this case, platforms like Grasshopper offer the user-base to encourage people to implement these methods for it.

This is also the case for more general purpose stochastic solvers. Grasshopper has access to both a Genetic Algorithm (Galapagos) and multi-objective NSGAII solver (Goat), both of which the author has applied on projects. These integrate well in parametric systems and are prime candidates for 'Innovisation' applications. Whilst favourable with respect to effort and time required to apply, especially as compared to self-built implementations like in the Bangalore example. They prevent more widespread automation processes being implemented, and so suffer from the same drawbacks of not using meta processes like being slower and less modular.

It is also argued that brute force searching is possible in many typical parametric systems, especially when parameters considered only those that are genuinely able to vary and have a meaningful impact on the design at its current stage, reducing search dimension bloat.

Visualisation

With systems able to produce larger quantities of data and faster than before, interfaces to understand and manipulate this data to gain insight comes to the fore. This is also being supported by better methods to visualise and understand complex data [Victor, 2011], [Bostock, 2013].

Work on Thames hub has offered much insight and feedback on how designers, clients and key decision makers consume technology. The most effective methods have shown to be interactive and easy to access on many devices. The webbrowser excels in this regard and it has become as ubiquitous in the developed world as a pen and paper. Importantly, it can be curated to filter and intelligently select relevant data, but also can be used to 'dump' data on to as an aid to understanding, as was applied in the Tocumen study.

These methods lower the complexity for interaction to just the web-interface; a medium where interestingly, the user is less apprehensive about using it as getting it wrong is blamed on the website rather than the user's actions as it is assumed that no specialist knowledge is required, unlike Excel.

These methods have also been successfully supported by modern programming styles of Python and JavaScript, to act as the glue between raw output data and polished user interfaces. There is an added benefit that this is also the language that the meta-processes were implemented in.

5.3 A Re-Combinable Set of Tools

The approach outlined above selects for processes that are chain-able, reconfigurable and prefers simple, preferably human-readable data formats like CSV and JSON, where possible. This comes from Unix and modern web development infrastructure; which has a strong sense of modularity, preferring tools that do one thing well, but which are easily connected together. This data driven approach is not just in-line with concepts of the current web but also the principals outlined for the 'Semantic Web' [Berners-Lee et al., 2001]. This is a proposal to change the emphasis of the web away from serving just documents, but also provide an infrastructure to intelligently enable views, operations and computing on relevant data, weather that is by computers of machines. This paradigm also highlights accurately what is to be server and client side by which there is a distinction between what is important to present well and what is only data is process.

The work presented has been carried out in this way, as a set of modular design assisting interventions that can be recombined, based on the task. However, it is of value to critically assess this approach to see how these methods can be combined as a whole and how this performs when used in practice. The opportunity to impose a comprehensive end-to-end implementation of an integrated performance driven processes was relatively difficult. This was in part due to the piecemeal nature of design studies, as well as the unintentional siloing of skills, like engineers expecting to do analysis whilst ARD group provide complex geometry support. As such the trailing of this process, required the buy-in from the engineers to allow a freer interplay between the disciplines. These challenges aside it was possible to apply these technologies on one project at an early stage in its entirety, and this will be explained below.

Figure 5-6: Exterior view of the proposed Cleveland Clinic as shown to press in November 2014. Source Foster + Partners.

Figure 5-7: Interior view of Cleveland Clinic main atrium, later version shown in November 2014. Source Foster + Partners.

5.3.1 Performance Driven Design Case Study

The Cleveland Clinic project was identified as an ideal project to use new design technologies on. There was more time available for design development than typical projects. It was also decided early on that there would be a large atrium with a feature roof. This roof would have to perform well structurally, but also be required to respond to environmental needs such as shading as well as architectural and aesthetic demands.

This highlighted the need for close control and understanding of the form. The structure represented a 168 by 100 meter void to span, which was substantial, especially as initially a light-weight roof solution was desired. The site supported this decision as there was low seismic activity in the area, thus, the dominating load case was imposed snow related, making the design less complex than highly dynamic structures undertaken elsewhere.

This reduction in constraints opened up the possibilities for creative designs. However, at an early stage, it was felt that a restrained simple structure would match the context and proposed building. There was also a desire to have a roof that was sufficiently shallow, so it would not be visible from the outside.

A simple gridded structure was proposed and chosen to progress. This was modelled parametrically, with initial design sessions with the engineers used, to set the sensible bounds on the grid size roof; maximum and minimum height, potential section sizes to consider.

A mathematical function was used to give the height of each of the grid's nodes. This function involved both centenary and sinusoidal terms, so as to promote efficient in plane load-take down, whilst not unduly rising too high, too quickly around the edges. The griddling nodes were evenly spaced in plan for simplicity.

Some analysis was undertaken by the engineers on basic centre-line models.

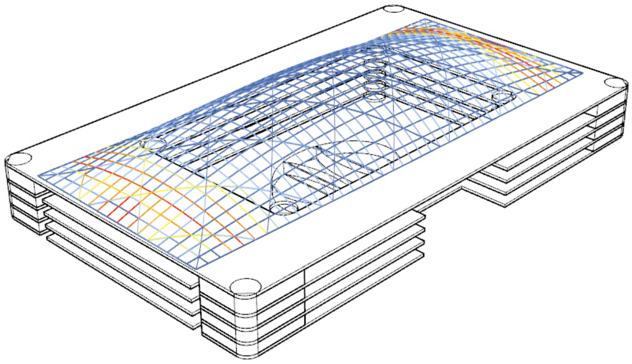


Figure 5-8: Roof stresses generated in a GSA analysis shown during live design session in 3D CAD software Rhino.

These proved the roof's concept, however, more tuning was required. The parametric system was then linked up to GSA, the structural software using the structural hub. This reduced model creation from 20 minutes, not including model transfer time, to less than a second.

The hub was used to return stress and forces data back to the parametric design environment, Grasshopper. This was then visualised by applying colour plots directly to the geometry of the elements concerned; crating a feedback loop from parameter change to analysis which was almost instantaneous. In this way, it was possible for both engineer and designer to see that the shorter edges of the plot were working significantly harder under uniform bidirectional grid stiffness.

This was useful not only for engineering understanding of the system, but was also useful in discussions with the architects, which could be presented in a CAD interface they are familiar with, and easily converted into models for rendering to make these arguments more compelling at wider design team meetings.

The analysis did show that the relationship between height, and grid dimen-

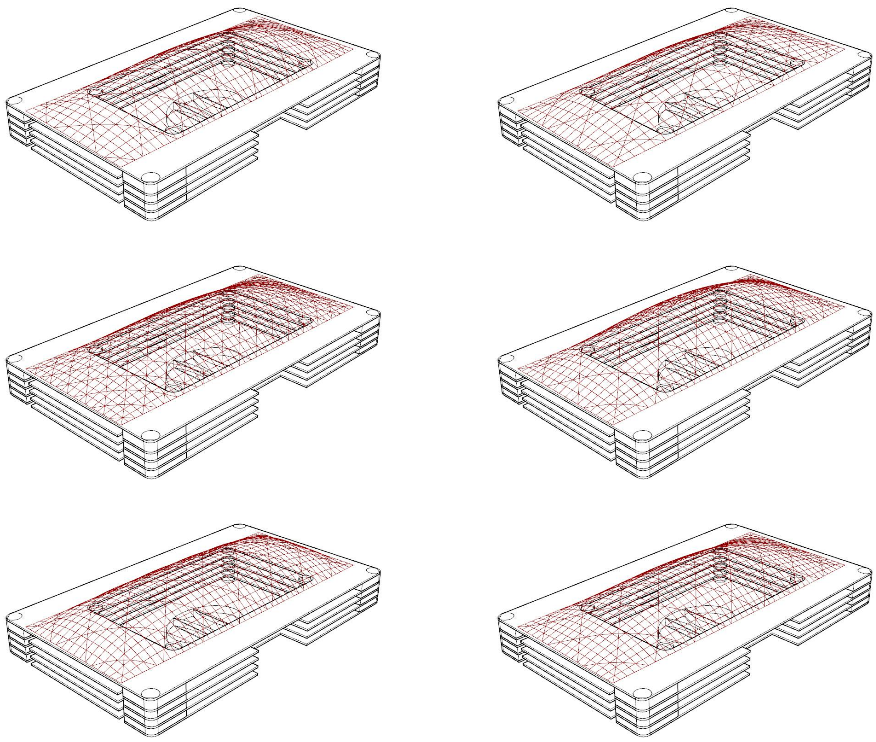


Figure 5-9: Some of the various roof variations generated

sions was not fully understood. This was made more complex when considering the cross-sections to be used to span both ways. The typical search approach would have been either to test a few options based on expected behaviour testing if the systems understanding was right, and then tune the values, or vary one parameter whilst keeping all others static based on reasonable assumptions, potentially or apply more sophisticated versions of this such as design of experiments. However, it was proposed by the author that we look into producing results for each possible variation altogether, to map the design and functional spaces together.

A study was generated to look into the effects of the different combinations of grid sizes, as well as roof height. Each one was analysed with all the key metrics saved as well as each individual model for later inquiry. Quick data analysis was able to show that the grid size was less significant than roof height within the parameters set. By showing that for any combination of grid a change in height improved it more than a change in either of the bay lengths.

The problem had more operational parameters however, and it was of value to know what were the most significant to obtaining optimal solutions. The principal parameter dimensions considered at the time were roof height, the bay length in x and y and sectional diameter. The design space was coarsely divided along these dimensions with 3, 4, 4, 3 different combinations for each, respectively. This generated 144 unique different designs.

A meta-process was developed to automate the generation of them which took less than 30 minutes to produce. Key analysis data, such as maximum loads as well as strain energy density of the elements as a measure of utilisation were stored. The volume of material was also recorded as this was the best early indicator to the design's cost. This provided a rich five dimensional design space, with an equally rich functional space. All of this was stored per analysis as a JSON file and these were aggregated together, to form a single data file.

The analysis of the data allowed a detailed Pareto front to be generated based on the comparison of internal stresses and the material volume, and see what where the configurations for the best trade-offs. This visualisation was enabled by the web visualisation tools, consuming a generic JSON file and allowing any numeric filed in the dataset weather input or output parameter to be graphed.

It was then decided that the view was incomplete, as the use of diagonals on the roof was not used. However, it was possible to modify the meta-process and run just for the new options to produce data for two diagonal density versions. These where appended to the original data set consumed by the visualisation. This now represented 432 unique models.

Although performance trade-offs were of value, it was also of value to understand the relationships between the functional performance of a model and its basic input parameters. Dimensions were introduced into the scatter plot to show the input values, to try to infer relationships. At its most complex, three extra inputs were displayed and this helped reveal what were the good combinations of

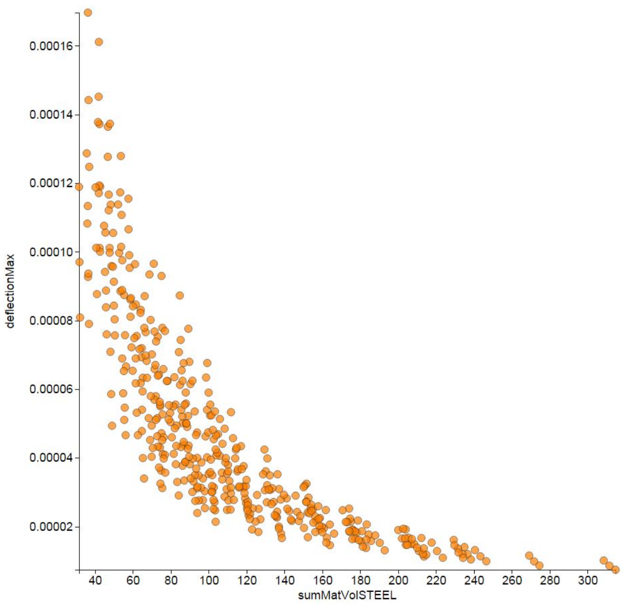


Figure 5-10: Pareto trade-off between structural weight and maximum stress

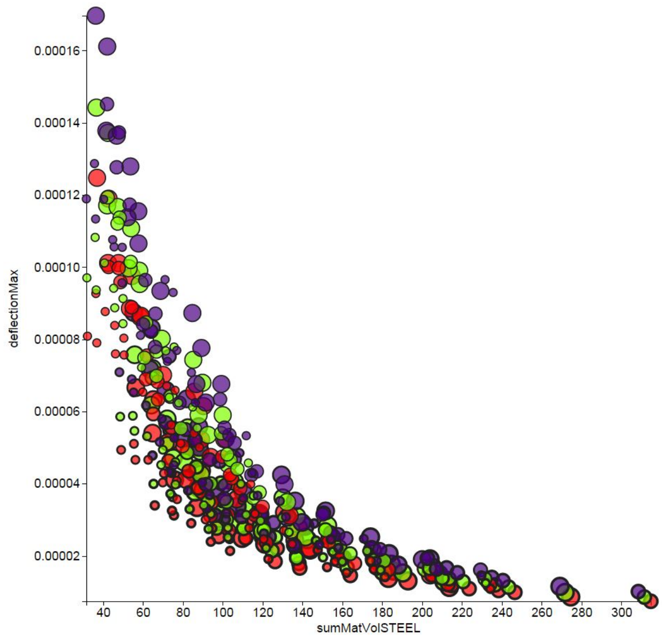


Figure 5-11: A version of the scatter plot in 5-10 with extra dimensions. Structural stress against material volume. Extra dimensions show the diagonal bays used for the point diameter, the height as colour and the stroke of the point circle as the section area.

different parameters to obtain the kind of performance required.

These plots were useful to arrive at high performing options, not just for picking the optimal versions but also having the flexibility to observe neighbouring options that were preferable for reasons not included in the analysis. However for understanding and explaining trends, these methods began to be too visually complex to read quickly, especially when explaining to others at speed.

Due to the generic nature of the data, it was easy to apply different plots. One example that was useful, was the parallel coordinates visualisation method. This provided a compact way to visualise the many dimensions. However with some modifications to the base code, it was possible to include selection sets over each of the dimensions, both functional input parameters and performance outputs.

This approach was applied reasonably successfully to understand and tune a number of options as the design progressed. This also included contemplating exceptionally thin arches using a similar geometry as Luigi-Nervi's 'Gaussian Arches', which have a 'S' type cross section in the mid span and then flattens towards the supports, to minimise buckling. This integrated CAD-analysis-data work-flow allowed the 3D printing of the buckling results, for explanation of the scheme.

5.4 Reflections

Towards the end of this study, the author's time was required on other critical projects and thus, this approach was not carried on deeper into scheme design as wished. The resultant roof design evolved into a more conventional single spanning truss system due to requirements from services and shading. However, the initial exploration was deemed useful by engineers and architects alike, with the application of a 'full stack' of the design exploration support tools, presenting a

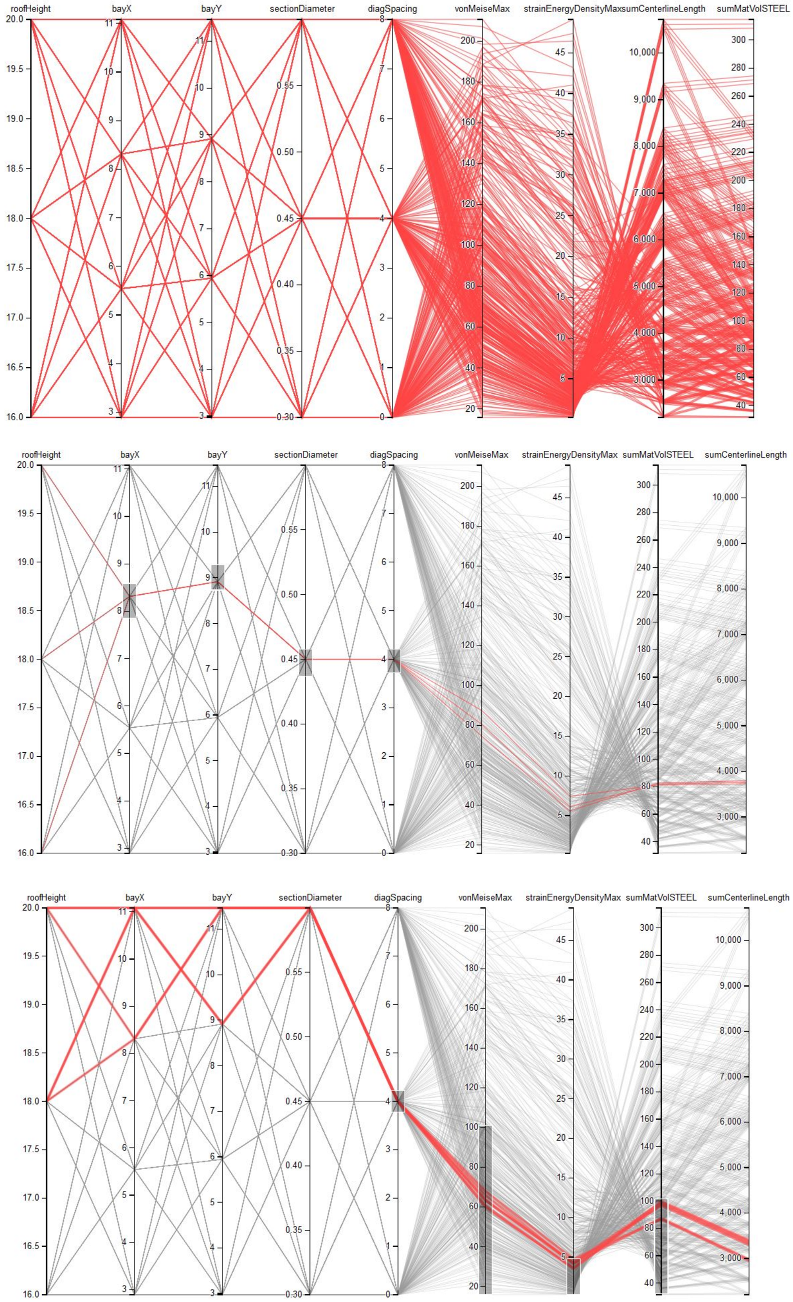


Figure 5-12: Example design session with Parallel coordinates web interface. Top shows complete design space with input variables on the left and performance metrics on the right. Second image shows a partial selection of some input parameters highlighted along with the resultant performance values. Third shows a performance criteria based selection highlighting the valid input values for those values.

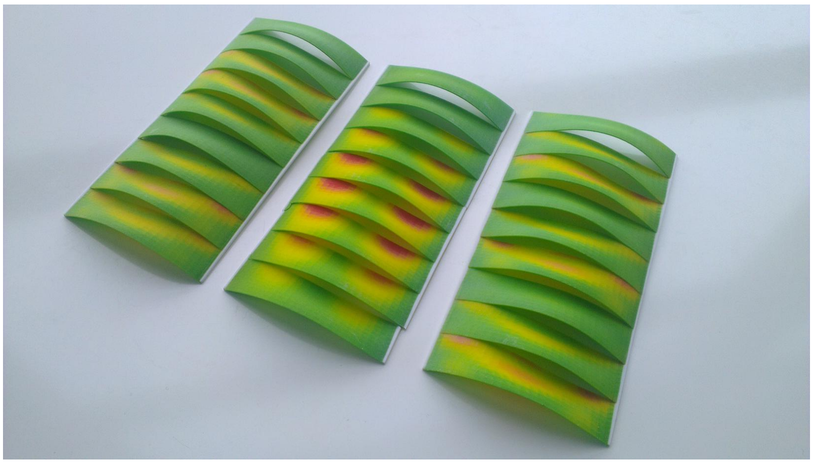


Figure 5-13: The modal shapes of Gaussian arch type structures, as colour 3D printed models.

novel way of undertaking integrated design in practice.

The fast generation and update of models both geometric and structural, that was enabled using parametric modelling and the structural-hub appeared to have the most impact; cutting iteration time down from a day to typically less than a minute, which sped up design evolution and the decision making process, although there was a challenge in convincing the stakeholders, primarily the engineering and design team to participate. However, after this, there was little reluctance in handing over geometric control as long as the reaction to design changes was relatively fast. In addition for relatively sophisticated geometry, the engineering team were wiling to allow the system to define loadings and run analysis as this would save time, as long as it had been verified manually in a few instances.

The more sophisticated uses of large scale model production were a harder sell; The typical way of exploring design options was very much engrained. The conventional approach which involved looking at a single model, discussing alternate options or a range and then looking at these had a natural progression to it. Where as, computing every example during a session presented challenges in determining what happened next.

Typically, these data analysis sessions required more time to understand and interpret the results. In part because comparing so many options was an alien process but also because it requires inductive reasoning to develop a mental model of the link between the performance results and the design space.

The use of this method also highlighted the assumption that the solution is in the data, may be false. In a conventional study, the emphasis was on the analysis model with a few results and plots to support this. Thus solutions proposed often related to changes of the geometry model. However, with a more data driven approach, solutions were arrived at by proposing a new value for a parameter, negating the possibility to change the model directly. Whilst this might be acceptable in confined design spaces in more early stages, this could be limiting.

5.4.1 Design Process

The case-studies covered present and argument for a high level of task switching is required between production, analysis, data/design understanding and decision making during the design process. With many of these transitions, they represent a change in task and software platform, as well as potentially person or persons involved. The diagrams in figure 5-14 attempts to capture some of this. These case-studies are important as they indicate some of the main process patterns found in practice, highlighting how modifications to these improve the process.

Tocumen

The original process represented a classic iterative tuning of a design which is undertaken both in engineering and architecture; The manual nature of building a model, analysis, extracting data, appraisal leading to feedback, making changes and starting the process over anew. Compare this with the alternative, which processes, on mass, many alternatives; This reduces the number of decision making

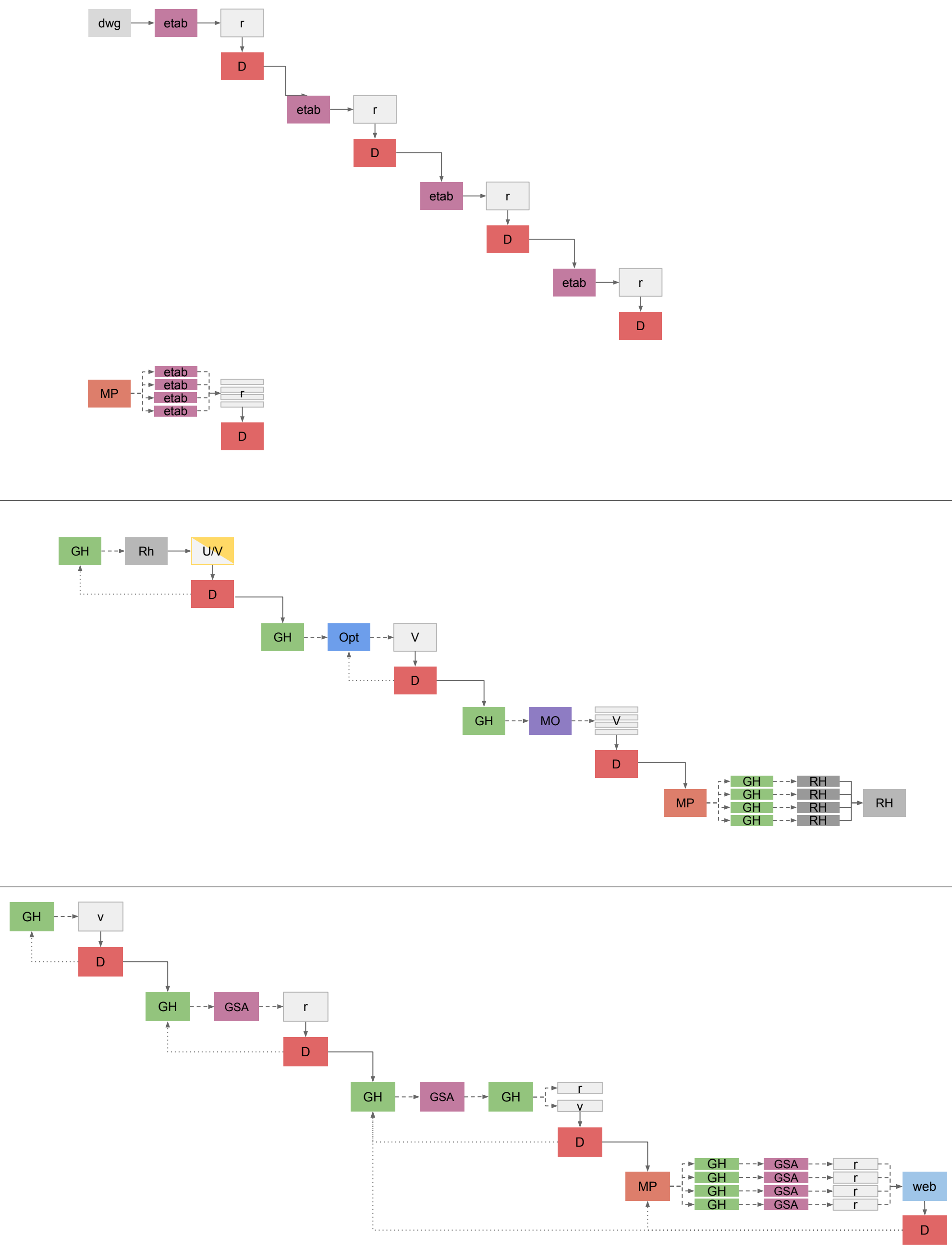


Figure 5-14: Process diagrams showing the transfer of data between packages and mediums. Showing in order Tocumen original and new, UAE Pavilion and Cleveland respectively. As well as solid, dashed and dotted being manual, automated and feedback translations. D decision process, v visualisation, r analysis results, MP Meta-Process, Opt Optimisation, MO Multi-objective optimisation, RH Rhino, Gh Grasshopper.

steps as more data is available in the first instance. In this case it is all the possible outcomes, shortening the effective decision making, into one step.

UAE

The UAE pavilion is indicative of a complex geometric project where difficult trade-off decisions have to be made, but within a shot period of time, thus here steps were taken to use method which widened the design exploration but resulted in singular solutions as supporting a range of options was not possible. This project also shows the evolution of performance driven decision making tools, over the duration of the authors involvement; Initially parametric modelling is enough, and helps create a fast feedback loop of parametric form definition, followed by modelling for visualisation production. However as more constraints are introduced, optimisation is required to get the best version. In cases with conflicting constraints, multi-objective optimisation helps to give a range of options so a reasoned decision can be made. Finally, at document production, a large amount of data is required, and by applying a meta process much time and effort was saved by automating and parallelising this.

Cleveland

Cleveland shows in even more detail how the complexity of meta-process can be built up over the duration of a design study. With simple parametric geometric definition, leading more integration with analysis, and then visual of that analysis. With the design generally acceptable, a large scale study of the parameter and objective spaces can be undertaken with much more data produced, which decisions informed by that. Leading to changes in design requiring more initial checking followed by wider analysis or by selecting a option.

5.4.2 Process Patterns

Combined, these examples work to exemplify that the conversion, analysis and decision making have a large impact on the amount of effort and time required. Decision making, concluding with a requirement to produce more options, not only requires people to rebuild models, reimport and re-analyse, but also requires a follow up meeting about that issue to ensure it is resolved. This all leads to significant extensions of time. By enabling a much wider set of options to be available, even if not used during a process, as well as the associated analysis data, the decision-making process can be made more potent and final.

Observed more widely, this analysis highlights certain reoccurring themes, with respect to work-flows. These could be linked to some of the strategies developed for parametric modelling by Hudson [Hudson, 2010]. In this work, Hudson outlines five main strategies:

- **kDev** Knowledge development strategy
- **kCap** Knowledge capture strategy
- **mCon** Model construction strategy
- **dInv** Design investigation strategy
- **cDoc** Construction documentation strategy

The 'kDev', 'dInv', kCap, 'cDoc' especially relate to specific periods in a design, and describe what tasks are undertaken during the different stages of development.

The case studies in this thesis correlate to these structures, with the case studies exhibiting applied methodologies to; boundary defining, directed exploration, decisive search or document production. For example, the last stages of UAE represent a ‘cDoc’ strategy or the Cleveland project being a strong case of an applied ‘dInv’ activity, and web option viewer being ‘kCap’.

The strategies capture systematised approaches to solving problems. Equally, the case study process diagrams capture the solution methods devised to solve the issues. In this way, they may be thought of to represent ‘Process-patterns’ examples of effective methods within larger systems. This is in a similar vain as ‘Design Patterns’ by Woodberry Aish and Kilian, which have been used to capture generic solutions of applying parametric modelling at a tactical level [Woodbury et al., 2007].

Chapter 6

Conclusions and Recommendations

"We shape our buildings; thereafter they shape us."

Winston Churchill, 1943

6.1 Conclusions

This thesis was wide ranging in its area of interest between architecture and engineering and computing and equally broad in its initial goals. The inquiry has gained insights into how engineering and design integration is currently undertaken, and proposed how they can be brought closer together via computation. It has developed new technological innovations, which importantly, has had a direct impact on professional praxis.

Unlike conventional research, the EngD differentiates its reach by being very much embedded in practice and the support of live projects. Whilst this enables a unique insight into the practical struggles and needs of industry, it also poses a challenge to implement research effectively. The concept of repeatability and controlled conditions to test ideas was simply not feasible. Fortunately, with Foster +

Partners being such a large organisation, there has been the opportunity to apply different approaches to similar problems, as well as hone and test these developments based on previous feedback.
To summarise what was covered in the chapters:

In Chapter 2, the basic infrastructure of modern computational design systems was surveyed, with respect to its impact at Foster + Partners and wider commercial practice, including current methods of integration between engineering platforms and CAD or design platforms; Identifying observed deficiencies in existing approaches and, implementing a new platform agnostic method to resolve them.

Chapter 3 focused on how we, as engineers and designers, go about rationalising and optimising their designs; presenting the current state of the art for geometric and structural approaches. These theoretical approaches were then contrasted to their actual application by the author on various design projects, Demonstrating that none of these provide a single solution but represent a step in a complex chain of process that contribute to an end result.

In Chapter 4, the application of meta-process was investigated with respect to computational design and analysis. Details of its development to provide scalable means of automating larger design tasks in a natural way were shown, as well as project case studies to build large 3D models and aggregate data on numerous model options. Following this were approaches to visualise and gain insight from data, responding to a demand to understand the large data sets produced.

In Chapter 5, a review of the previous studies was used to uncover the patterns in the processes. The role of search and exploration was found to be key, highlighting that many existing engineering approaches to design are centred around being conclusive, but do not support methods of broadening the design space. A response to this was shown as a case study, where the aforementioned tools were chained together with the aim to enable better exploration and understanding whilst still focusing on functional performance requirements.

6.1.1 Contribution To Knowledge

Whilst the work has had primarily an applied industrial focus, there have been a number of contributions to academic knowledge over the long duration of this research. Over 9 peer-reviewed papers in have come out of the work, including 6 published during the period of involvement with Foster + Partners. Two papers have not been included as it was viewed not relevant to the thrust of EngD research.

These covered papers show the breath of approaches investigated. Covering a range of technical developments for practical application of form finding an optimisation approaches [Joyce et al., 2011], [Evins et al., 2012], [Maleczek et al., 2013], [Williams et al., 2014] , as well as case study examples of these approaches applied to projects [Tsigkari et al., 2013], [Malm et al., 2015] . There is also a body of publications which represents investigation into methods of computational design and shows growing focus on open design process [Harding et al., 2013], [Aish et al., 2012] and data visualisation [Joyce, 2015].

In many ways this work traces the arc of the research interests and findings over the duration. Coming from technical 'hard' solution origins where computation solves specific identifiable issues but ones that are often isolated and separated from the human 'soft' side of design processes, potentially removing control and insight from the designer. This work shows a general trend moving towards a more collaborative stance which seeks to support designers by process which readily provide performance data and still providing choice.

6.1.2 Joined Up Thinking

Whilst arguably the theory and technology used and developed in this thesis are not new, it is proposed its synthesis is. It has been the observation of the author that with many problems in a real design context, it is rarely the running of solvers or algorithms used that represents the largest use of time but how they are coupled together. The discontinuities between engineer and architect, optimisation and design appraisal, CAD software and FEA software, present the largest hurdles to working together efficiently.

Some have proposed that this can be solved by centralising all of this data in one system such as BIM or Vasari for conceptual development. The author has been involved in the creation of relatively monolithic solutions [Aish et al., 2012] in this case for Autodesk. From this experience it is felt that these efforts, most suit software companies, over their users, by locking them in their closed technology ecosystems. This is opposite to what has been experienced in practice, where the best projects come from talented but diverse people collaborating using the best tools for their individual task. Thus are more open and inclusive rather than exclusive ecosystem is more likely to integrate the best software tools for the job.

Presenting a single system solution requires the developer of said system to pre-empt and code all the design, analysis and visualisation needs. However, invariably designers need something that wasn't thought about by software engineers, and are in fact compelled to probe into, and break from the limitations tools provided in order to obtain a unique result.

To this end, it is believed that much can be learned from the loosely coupled technology-stacks of internet infrastructure. In this model, data is typically separated from analysis, which is separated from visualisation and user interface. The primary emphasis is not on the tools but how they are glued together to make a system. This has the advantages of being more flexible to change, as well as, potentially much more scalable with modern software and hardware.

This thinking is what has produced the design interventions, outlined in the thesis. The aim was to produce tools which would support performance driven design, and this was necessarily, broadened to include design understanding, and

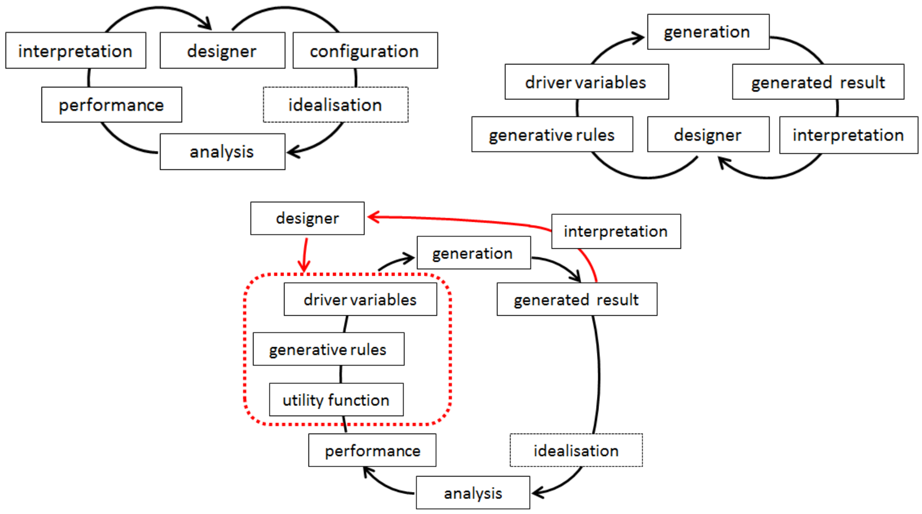


Figure 6-1: Proposed evolution of design analysis and evaluation process as laid out in [Aish et al., 2012].

exploration. To do this practically, the work used improved data flow between programs, scalable controllable meta-processes over existing work-flows and modern data visualisation methods.

As represented in figure 6-1, the use of parametric tools acts to abstract the designer from the design. By requiring them to act on the design generation logic which produces a range of options rather than one singular artefact. This, whilst offering new levels of flexibility to explore, also has the danger of leaving the modeller in a constant state of undefined pluralism. By introducing methods of producing performance metrics in keeping with the open degrees of freedom that parameters allow, coupling this with sophisticated visualisation, helps the designer to interpret what they have produced and enact changes.

6.2 Recommendations

This research has developed a few key recommendations, summarised here:

- The parametric modelling paradigm represent useful design intent capture systems, and enable much wider search of design space by enabling values to be changed. However, they are not a singular solution and need to be thought of in a wider context of designers and engineers.

- Barriers to transferring between geometric and analysis models must be reduced with processes automated to enable metrics to be integrated into design decisions.
- Design is a complex constant reconfiguration of lightly coupled dependent activities. Computational approaches must be modular and amenable to restructuring to adapt to this.
- Developing processes that are fully automated and scalable, allows for wider searches later on.
- Design represents a sliding scale between exploration and search, representing a broadening of the design space and a selecting of specific preferred options respectively.
- Exploration and search actives happen at different times of the design process with exploration earlier and search latter.
- Computational assistance of search and exploration can be realised using techniques such as optimisation and data-visualisation respectively.
- Visual understanding of design spaces leading to designer understanding of the performance values and how they relate to design variables, are more useful than simply finding the optimal solution in many design instances.
- The screen is a medium that does not have to mimic paper and can utilise new possibilities such as interactivity and transitions, to increase understanding and informational bandwidth conveyed.
- The decision making process is of critical importance; good sessions require a wealth of data not only of one option but of a range of potentials. This min-

imises uncertainty and/or need to iterate options leading less duplication of similar work and more effective design time.

6.3 Criticism

What has been proposed has proven to be useful in projects but is by all means not complete. The main criticisms are believed to be in two areas; the applied methodology of action research embedcded in practice, as well as the proposed recommendations of interconnected performance centered tools configured to suit requirements.

6.3.1 Methodological Criticism

Project Centric Approach

All the of the work highlighted has derived from project needs as determined either during project delivery or on reflection after completion. In this way, these interventions potentially suffer from having to be constantly of value to a project or in danger of being sidestepped or ignored during a process. The position held by the author, a gatekeeper in the geometric and structural definition of many of the projects discussed, leads to a level of control to steer project interests. However care has always been exercised not to needlessly focus efforts of a project down an academic route of study, when others require critical deliverables to continue their own involvement.

It is believed that in some instances, a more remote position from project needs would have enabled a more complete and resolved approach to be devised. The author by being relied on to constantly contribute key deliverables of a project whilst at the same time devising strategies to pose and implement solutions to generalisations of these problems, has at times been to the detriment of both. This being said, the role of research and development working within a practice, and having to prove their worth on real projects is a condition that seems to be of rising prevalence. As such this research, by conveying how and how-not, the author has been able to implement effective design processes within an organisation via technological change; and it is believed this is an important contribution as the solutions themselves.

Measuring Change

With design activities, there is a difficulty in the measurement of 'better' approaches. This is the, all creative endeavours, as the creative approach is not repeatable, and thus, different sessions are not comparable. This is even more so for a practice like Foster + Partners, as clients approach the firm looking for iconic design, which almost by definition, requires something that has not been designed before. This is equally true within the firm, which although accepting inspiration from previous and existing projects, will actively try to search out a different approach to synthesise a new solution. This makes objective comparisons between current and the new methods difficult.

In some cases, it has been possible to observe and measure processes before and after intervention. This is especially the case in instances where the new approaches directly replace a pre-existing approach. In examples such as the Madrid Stadium Roof GSA model creation, or the UAE panel modelling, it was possible to benchmark existing manual processes with the new ones, to measure relative differences in time taken and performance of output. However, it has been shown that the design exploration is arguably the more impactful stage, but this is also where comparisons are less easy to obtain.

For some of the case studies, feedback from others was used to understand and compare. However, in instances where a process was changed, a comparison, other than qualitative is not possible. For examples where a large range of designs was used, it is believed the designer's behaviour was modified, allowing them to take on more options and data during each session leading to a wider search. It is believed that this resulted in a better design outcome. However as it is unreasonable to get the same people to redo the design, without assistance to compare the difference, and even if this was undertaken, it would now be with prior knowledge of the problem, skewing the 'independent' result.

Ultimately, the quality of the building is the final test of the quality of design-thinking and integrated process. and it is hoped that in time, projects have been completed using these techniques will be appreciated.

6.3.2 Proposal Criticism

An Ethos Rather than a Solution

The highlighted approach represents recommendations and working examples of modular applied approaches but ultimately, does not offer one canonical solution. Although it is a criticism that a more all encompassing solution, to producing high performance designs was not arrived at, it is also an important finding. It has been the experience of the author that no one way is right, and that forcing an approach can lead the design process to a dead-end, or make it resist the application of new approaches. Instead, by assisting the design exploration, augmenting the process with performance metrics and enabling better understanding of those metrics, these methods add to the design journey and become accepted.

Understanding: Deduction vs Induction

The data analysis component of the proposed approach is important to derive understanding and decision making from data. This requires inductive reasoning if any options outside of what has been generated are to be proposed. This is important to allow the development of the design. However, owing to the large dimensionality of the data, it is not as trivial a test as it might seem, when looking at basic single variable changes in parameters and observing the change in its objective performance. Whilst it has been observed that experienced designers and engineers are very capable of interpreting important relationships from large datasets, in the Cleveland case study, this was pushed to the limit and it was found to be a struggle at times. The volume of data coupled with the number of dimensions represented, made it difficult in some cases to effectively process and glean useful information out of this. However this is an important facet for the exploration of options.

It is of course possible to reduce the amount of data displayed, creating slices or generating a reduced set. This reduces some of the power of inductive understanding of how the parameter changes result in performance changes. In time, users found it easier to interpret results from complex data visualisation systems. However, there is definitely an upper bound on how much data can be meaningfully understood.

Implementation Complexity

It is the case with the current proposal that still a relatively high level of technical skill is required to implement these ideas. Skills applied include parametric modelling, integration of different software by programming plug-ins, building meta process and knowing about networking to parallelise them as well as developing custom website designs. These skills require a relatively high level of technical capabilities, on top of existing design or engineering knowledge. This is currently true, but developing interest in parametric platforms within the wider design community leading to support for and development of more user friendly tools. Software such as Grasshopper, Geometry Gym, Google Graphs already provide easier more open interfaces for those less capable and this trend looks to continue.

It is the view of the author that in the short-term, this will still require someone to own and support computational processes. This new role in design has been explored in detail by others [Hudson, 2010], referred to as 'parametric-designers'. However, the author believes that this role is diversifying to include not only parametric modelling but also wider computational and process integration. This is possible due to the improved usability of parametric tools, reducing workload in this regard, but also allowing more time to widen the influence of computational processes. Furthermore, the author has observed that more basic parametric design is undertaken by younger architects, leaving those parametric-designers to tackle the harder complex tasks as outlined.

The Limits of Exploration

It must be appreciated that there are practical limits to the area explored. In a basic parametric sense, each parameter represents an increase in dimension of the search space. When calculating the number of discreet options each dimension adds one to the exponent. With this, exponential increase soon being preventative to realistic exploration, irrespective of parallelisation at least if we exclude considerations of the future of quantum computing.

6.4 Future Directions

This research has uncovered more problems than it has answers. Sadly, this research's own inductive approach means that many problems are identified and well-formulated only at the end of the work. As such, there are a range of new but complementary directions that could be perused, and this section offers the opportunity to detail these.

Process Standards

In his thesis "Modelled on Software Engineering: Flexible Parametric models in the Practice of Architecture" [Davis, 2013], Davis makes the argument that the creation of parametric models could benefit from learning from the strategies and mistakes of the software industry. After experience of practice, Davis' emphasis on using well structured parametric models to encourage greater reuse of design effort, is certainly echoed by the author.

It is the author's belief that this thinking needs to be taken further outside of parametric models, and applied to computational design processes more generally. Despite its growing reach, parametric modelling will never encompass all requirements in design, not least engineering analysis which requires its own specialist knowledge and skills. The detailed modal analysis, as well as, data visualisations are both examples, neither of which are ideally suited to parametric systems, but data can be usefully generated and processed by them.

Design processes would also benefit from a level of standard interfaces and accessibility to ensure that new processes could be quickly and predictably made, by aggregating systems together. Software already has a few well-used basic information processing protocols to enable this in its domain. At its most basic, human readable text or JSON data exchanges provide a low barrier to integration. The function with its basic arguments and return values has been very effective in generating processes especially when coupled with a Unix like process, managing and data piping environment.

If there was more heterogeneity with respect to the high-level control and interfacing, then this would help immensely in setting up processes quickly and efficiently. The Python programming language is becoming something of a standard in data processing and now slowly with Grasshopper, Rhino and Dynamo supporting it. However, effort to identify and define the standards required would be a worth while effort, but one, which only would be of value if it had major cad developers buy-in.

Version Capture/Control

During a design session, a geometry model is created which is liked but owing to an unknown changed, parameter or reconfigured logic is not easily retrievable; Even with careful saving and archiving parametric models, their outputs can become dissociated. One accidental finding of developing logging methods for analysis results was that a parametric model could be 'exercised', producing different designs with different parameters as usual, but the logging would capture all this data. This could then be used to navigate through different proposed options to bring them up and compare them. Benefits for decision making were found using the UAE version explorer, however the version retrieval was manual and time consuming. This points to the idea that version capture and control as is used in software would be of use.

Whilst the development of parametric models has similarities to software engineering, it differs in that the geometry model is being developed at the same time as the generic logic with which, to create it. To generalise, the data processing (model creation) is undertaken at the same time the software or process is being developed (parametric modelling). This is not the case with software development; software development is usually not interested in the specifics of the input data during development, as long as, the program functions as expected and the output result is correct. Thus, code versioning tools concentrate on the process and, not what has been put through it. This means that existing software development tools are not well placed to capture the processed data, as well as, the state of the existing system.

A more fitting usage paradigm is in the professional practice of analytical modelling and data processing. This is because in this field, the computational mod-

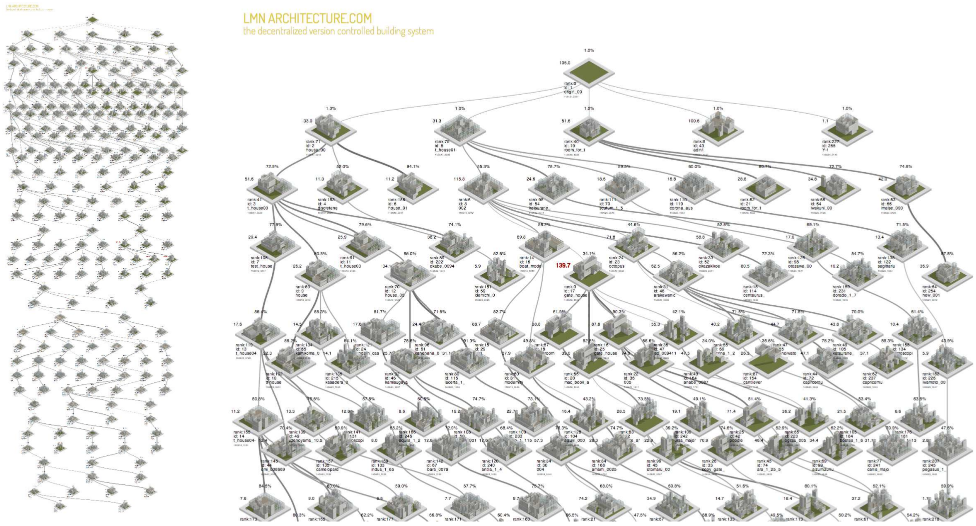


Figure 6-2: Version tree and close up from LMN Architecture from [Sakai and Tsunoda, 2015]

els are updated based on feedback from the data. However, the ability to recreate both the models and the data is important. One widely used tool is Luigi [Bernhardsson, 2014] which is designed to work with long-running batch processes, not unlike the multiple parametric-model generation work by the author. This tool however, is more generic and aims to help piping data to large scale cloud computing, managing the data dependency graph to update data processing down stream. It aims to save each batch's results along with the process data (code), and the library dependencies. If applicable to design processes this kind of support would be very useful in practice.

Interesting work in versioning or option development capture, has been undertaken by by Saki and Daisuke with 'lmn architecture system' [Sakai and Tsunoda, 2015]. They have developed a web interface to design a simple cell based design. Whilst the actual design variability is limited using a grid density area use, the approach to option capture is novel. Each new design is a variation of an existing one; a seed design is chosen from the option tree by the user, modified and then, can be committed back to the tree. In this way, every change can be captured and revised by the designer, storing meta-properties such as date generated or properties of the design itself.

In many ways this mimics the current trend in cloud based code version control. They promote decentralised collaborative working on code bases, by devising methods to manage changes; allowing people to fork and merge code bases, as well as, revert back to previous states. Whilst beyond the scope of this current research, this could have significant application in large offices such as Foster + Partners, if applicable to production design systems. By having a version control, general changes and design exploration would have less risk as the process could be reset to a working one, with all options stored and recorded for quick access later.

The 'lmn system' works in a web browser with its own 3D modelling interface and as such exists separately from the systems used by the majority of designers. As such they represent a promising proof of concept, rather than a working example, but this would be of value if applied to production design software.

Distributed Computing

The computing undertaken in this research has aimed to be as scalable as possible, and although in some instances it has been run over the network on separate machines, a much more powerful system could be created if it was able to utilise cloud computing. Cloud computing is where a large number of basic machines are available as processing nodes or even whole cloned computer instances. This has had significant impact in data analysis, with the provision of groups like Amazon, Google and IBM offering paid-for-access to their servers.

Current implementation issues are more complex for architecture and engineering as propriety software is required to run many design processes and so interfacing and licensing with this would represent a challenge. The research by the author has shown that meta-processes can be effectively used to control such design tasks, however it is felt that more work would have to be done to make this generally practical.

There is precedent for developing these kind of distributed systems in architecture; The ParaGen system [von Buelow, 2012] represents a server based method for computing of options from the multi-objective optimisation of a given GC parametric model. This benefits from the light weight text properties of GC models as this minimises network data transfer requirements when starting the optimisation. Equally, results returned represent curated values and model screen-shots, avoiding issues with large data stores of full 3D models. With models selected via a interface, which can then be built on the local machine if a full model is required.

Artificial Intelligence

Currently the existing methods rely on human design understanding or objective performance requirements, to select for and define trade-offs. This puts a strain on the power of understanding of the user. However, in many areas of social media and big-data analysis, computational methods of interpretation are being used. These methods are able to perform activities such as grouping similar objects, correlating peoples' personal preferences, and correlating changes in input data with results [Segaran, 2007].

All of these would be useful in preprocessing the large datasets generated, so as to lessen the initial effort on user's understanding. At its most basic, applying methods to help in regression analysis may be able to correlate input values with resultant model performance. These methods have grown in capability recently with techniques such as neural networks and support vector machines, repenting robust but powerful methods to produce higher-level descriptions of the relationship between performance and parameters [Cortes and Vapnik, 1995], [Haykin and Network, 2004].

A promising application of artificial intelligence in structural design has been demonstrated by Mueller, allowing users to provide their own subjective input on design examples, using interactive evolutionary frameworks as a way to 'learn' user preferences by combining subjective user requirements with performance metrics in the same optimisation and [Mueller and Ochsendorf, 2011].

More abstractly, tools such as Self-Organising-Maps represent methods to improve understanding of large dimensional spaces by dimensionality reduction; something that could help in spotting similarities in performance spaces and classifying them as different solutions more easily. Some of these methods were employed for the automatic identification of office space clustering described previously. This points to an approach which could enable high-level understanding of early fundamental design decisions, by allowing the A.I. approach to fill in the blanks reasonably.

Another, is to employ A.I. to intelligently synthesise designs. This approach is arguably already initialised using genetic programming on Grasshopper models. However, its low-level of interaction and difficulty introducing new requirements would need to be addressed, to have a real impact.

6.5 Conclusions

The work highlights the transformative effect of parametric systems on design processes, but also how the rest of the engineering and integrated design has yet to catch up and adapt to the new paradigm. Parametric modelling by moving emphasis away from simple direct representation, towards capturing design intent, presents new possibilities but also requires new thinking to unlock its potential. This is especially true in the case of the generation, analysis and interpretation of the functional performance of a design.

This work has investigated approaches to improve the production and integration of engineering, specifically in structural engineering metrics. By using parametric modelling as a standard to build design logic, developing methods to integrate analysis and meta-processes to computationally drive search, an eco-system of tools has been produced. This allows for more efficient methodologies to undertake optimisation and design space exploration by generating any range of options, supported with performance metrics.

The challenge of staring from nothing to producing an effective beautiful solution is the essence of design. The design process has been revealed to be significantly more than a simple case of optimising for requirements; it is a creative exploration activity. This is perhaps best summed up by Rittel and Webber:

> “Take an optimization model. Here the inputs needed include the definition of the solution space, the system of constraints, and the performance measure as a function of the planning and contextual variables. But setting up and constraining the solution space and constructing the measure of performance is the wicked part of the problem. Very likely it is more essential than the remaining steps of searching for a solution which is optimal relative to the measure of performance and the constraint system”.
>
> [Rittel and Webber, 1973]

This view is very much corroborated by the author’s experience of practice. Also identifying that the problem definition is very much effected by the prevalent metaphors used as well as commercial design culture which sets a philosophy of how a group of people search for solutions [Coyne and Snodgrass, 1995]. With the definition and creation of a solution space, dependent on meaningful feedback from existing proposals. The more information about a given design or design space, the better informed decisions can be.

These concerns are not new. In May 2002, the then fledgling SMG group presented to the wider Foster office, a talk entitled: “Exploring Solution Space : A metaphor for design; a term borrowed from Optimisation Theory”. At that time

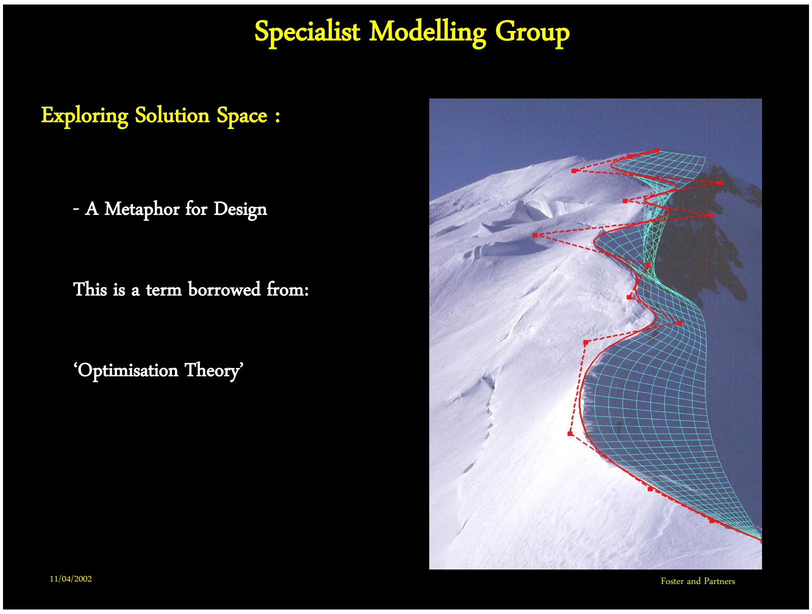


Figure 6-3: Early internal presentation of the SMG group to the office, entitled 'Exploring Solution Space' from 2002 [Whitehead and Josefsson, 2011]

the group consisted of only four people and the whole Foster + Partners office was made up of 600. Now the SMG and ARD groups combined, represent over 40 people with the office having approximately 1400 employees. This growth is indicative of the effectiveness of these approaches on design projects.

In that talk, although there was discussion about optimisation borrowed from engineering, much of the work was geometric. This is perhaps indicative of the challenges of the time, which were predominantly geometric. With greater experience of parametric design, and new software some of these challenges have been made easier; with this approach becoming more fundamental to how design is undertaken. However, in this way, new challenges of analysing optimization and interpreting these forms, have come to the fore, and it is hoped that the work described, has helped rise to this new challenge.

Bibliography

[Adamu et al., 2015] Adamu, Z., Emmitt, S., and Soetanto, R. (2015). Social bim: Co-creation with shared situational awareness. *ITcon-Technology Strategies for Collaborative Working*, 20:230–252.

[Addis and Walker, 2005] Addis, B. and Walker, D. (2005). *Happold: The Confidence to Build*. Taylor & Francis.

[Adriaenssens et al., 2014] Adriaenssens, S., Block, P., Veenendaal, D., and Williams, C. (2014). *Shell structures for architecture: form finding and optimization*. Routledge.

[Aish, 2000] Aish, R. (2000). Custom objects: A model-oriented end-user programming environment. In *Visual End User workshop, IEEE Visual Languages, Seattle*.

[Aish, 2003] Aish, R. (2003). Bentley's generative components: a design tool for exploratory architecture. *Bentley Systems Inc*.

[Aish, 2012] Aish, R. (2012). Designscript: Origins, explanation, illustration. In Gengnagel, C., Kilian, A., Palz, N., and Scheurer, F., editors, *Computational Design Modelling*, pages 1–8. Springer Berlin Heidelberg.

[Aish et al., 2012] Aish, R., Joyce, S., Fisher, A., and Marsh, A. (2012). Progress towards multi-criteria design optimisation using designscript with smart form, robot structural analysis and ecotect building performance analysis. In *Synthetic Digital Ecologies, ACADIA Conference Proceedings, San Francisco, USA, 18–21/10/12, pp 47-56*, page 10. ACADIA.

[Aish et al., 2013] Aish, R., Peters, B., and Peters, T. (2013). First build your tools. *Inside Smartgeometry: Expanding the Architectural Possibilities of Computational Design*, pages 36–49.

[Alexander, 1964] Alexander, C. (1964). *Notes on the Synthesis of Form*, volume 5. Harvard University Press.

[Allinson and Thornton, 2014] Allinson, K. and Thornton, V. (2014). *London's Contemporary Architecture: An Explorer's Guide*. Taylor & Francis.

[Ashby et al., 1956] Ashby, W. R. et al. (1956). *An introduction to cybernetics*, volume 2. Chapman & Hall London.

[Asimow, 1962] Asimow, M. (1962). *Introduction to design*, volume 394. Prentice-Hall Englewood Cliffs, NJ.

[Baker, 1991] Baker, W. F. (1991). Stiffness optimization methods for lateral systems of buildings: A theoretical basis. In *Electronic Computation (1991)*, pages 269–278. ASCE.

[Bärtschi et al., 2010] Bärtschi, R., Knauss, M., Bonwetsch, T., Gramazio, F., and Kohler, M. (2010). Wiggled brick bond. *Advances in Architectural Geometry 2010*, pages 137–147.

[Bendsøe and Sigmund, 1999] Bendsøe, M. P. and Sigmund, O. (1999). Material interpolation schemes in topology optimization. *Archive of applied mechanics*, 69(9-10):635–654.

[Bentley and Kumar, 1999] Bentley, P. and Kumar, S. (1999). Three ways to grow designs: A comparison of evolved embryogenies for a design problem. *Genetic and Evolutionary Computation Conference*, pages 35–43.

[Bentley, 2014] Bentley, S. (2014). Foster + partners competes for gold at the sse hydro in glasgow. `http://www10.aeccafe.com/link/Foster-+-Partners-Competes-Gold-SSE-Hydro-Glasgow/44031/link_download/No/FosterAndParnters_CaseStudy.pdf`. [Online; accessed 02-Jan-2015].

[Berners-Lee et al., 2001] Berners-Lee, T., Hendler, J., Lassila, O., et al. (2001). The semantic web. *Scientific American*, 284(5):28–37.

[Bernhardsson, 2014] Bernhardsson, E. (2014). Luigi a python based batch job pipeline manager. `http://luigi.readthedocs.org/en/latest/`. [Online; accessed 23-Feb-2015].

[Bhooshan et al., 2014] Bhooshan, S., El-Sayed, M., and Chandra, S. (2014). Design-friendly strategies for computational form-finding of curved-folded geometries: a case study. In *Proceedings of the Symposium on Simulation for Architecture & Urban Design*, page 19. Society for Computer Simulation International.

[Block and Ochsendorf, 2007] Block, P. and Ochsendorf, J. (2007). Thrust network analysis: A new methodology for three-dimensional equilibrium. *Journal-International Association for Shell and Spatial Structures*, 155:167.

[Bostock, 2013] Bostock, M. (2013). Data-driven documents (d3. js), a visualization framework for internet browsers running javascript. `http://d3js.org/`. [Online; accessed 23-Feb-2015].

[Bradner et al., 2014] Bradner, E., Iorio, F., and Davis, M. (2014). Parameters tell the design story: ideation and abstraction in design optimization. In *Proceedings of the Symposium on Simulation for Architecture & Urban Design*, page 26. Society for Computer Simulation International.

[Burger, 2008] Burger, S. (2008). Grimshaw architects. In *ACM SIGGRAPH 2008 art gallery*, pages 26–27. ACM.

[Burry and Gaudí, 2007] Burry, M. and Gaudí, A. (2007). *Gaudí unseen: completing the Sagrada Família*. Jovis.

[Ceccanti et al., 2010] Ceccanti, F., Dini, E., De Kestelier, X., Colla, V., and Pambaguian, L. (2010). 3d printing technology for a moon outpost exploiting lunar soil. In *61st International Astronautical Congress, Prague, CZ, IAC-10-D3*, volume 3.

[Checkland, 1999] Checkland, P. (1999). *Systems thinking, systems practice: includes a 30-year retrospective*. John Wiley And Sons.

[Checkland and Scholes, 1990] Checkland, P. and Scholes, J. (1990). *Soft systems methodology in action*. John Wiley And Sons.

[Chomsky, 2002] Chomsky, N. (2002). *Syntactic structures*. Walter de Gruyter.

[Chronis et al., 2012a] Chronis, A., Tsigkari, M., David, A., and Aish, F. (2012a). Design systems, ecology and time. In *ACADIA conference proceedings*.

[Chronis et al., 2012b] Chronis, A., Tsigkari, M., Giouvanos, E., Aish, F., and Zaki, A. A. (2012b). Performance driven design and simulation interfaces: a multi-objective parametric optimization process. In *Proceedings of the 2012 Symposium on Simulation for Architecture and Urban Design*, page 14. Society for Computer Simulation International.

[Chronis et al., 2011] Chronis, A., Turner, A., and Tsigkari, M. (2011). Generative fluid dynamics: Integration of fast fluid dynamics and genetic algorithms for wind loading optimization of a free form surface. In *Proceedings of the 2011 Symposium on Simulation for Architecture and Urban Design*, pages 29–36. Society for Computer Simulation International.

[Coates, 2010] Coates, P. (2010). *Programming. architecture*. Routledge.

[Coates et al., 1999] Coates, P., Broughton, T., and Jackson, H. (1999). Exploring three-dimensional design worlds using lindenmayer systems and genetic programming. *Evolutionary design by computers*, pages 323–341.

[Coates et al., 1996] Coates, P., Healy, N., Lamb, C., Voon, W., et al. (1996). The use of cellular automata to explore bottom up architectonic rules. In *Eurographics UK Chapter 14th Annual Conference*, pages 26–28. Citeseer.

[Coenders, 2012a] Coenders, J. (2012a). Networkeddesign, next generation infrastructure for design modelling. In *Computational Design Modelling*, pages 39–46. Springer.

[Coenders, 2012b] Coenders, J. (2012b). Networkeddesign, next generation infrastructure for design modelling. *Computational Design Modelling*, pages 39–46.

[Cortes and Vapnik, 1995] Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3):273–297.

[Coyne, 2005] Coyne, R. (2005). Wicked problems revisited. *Design studies*, 26(1):5–17.

[Coyne and Snodgrass, 1995] Coyne, R. and Snodgrass, A. (1995). Problem setting within prevalent metaphors of design. *Design Issues*, pages 31–61.

[Cremona, 1890] Cremona, L. (1890). *Graphical statics: two treatises on the graphical calculus and reciprocal figures in graphical statics*. Clarendon press.

[Darwin, 1871] Darwin, C. (1871). On the origin of species by means of natural selection. *Murray. London*.

[Davis et al., 2014] Davis, A., Hanna, S., and Aish, F. (2014). Characterising place by scene depth. *Poster Abstracts Design Computing and Cognition14*, page 17.

[Davis, 2013] Davis, D. (2013). *Modelled on Software Engineering: Flexible Parametric Models in the Practice of Architecture*. PhD thesis, RMIT University.

[Davis et al., 2011] Davis, D., Burry, J., and Burry, M. (2011). Untangling parametric schemata: enhancing collaboration through modular programming. In *Proceedings of the 14th international conference on Computer Aided Architectural Design, University of Liege, Liege*.

[Deb et al., 2002] Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: Nsga-ii. *Evolutionary Computation, IEEE Transactions on*, 6(2):182–197.

[Deb and Srinivasan, 2006] Deb, K. and Srinivasan, A. (2006). Innovization: Innovating design principles through optimization. In *Proceedings of the 8th annual conference on Genetic and evolutionary computation*, pages 1629–1636. ACM.

[DeLanda, 2002] DeLanda, M. (2002). Deleuze and the use of the genetic algorithm in architecture. *Architectural Design*, 71(7):9–12.

[Derix, 2009] Derix, C. (2009). In-between architecture computation. *International journal of architectural computing*, 7(4):565–586.

[Derix et al., 2011] Derix, C., Kimpian, J., Karanouh, A., and Mason, J. (2011). Feedback architecture. *Architectural Design*, 81(6):36–43.

[Dréo, 2011] Dréo, J. (2011). Metaheuristics classification. http://commons.wikimedia.org/wiki/File:Metaheuristics_classification.svg. [Online; accessed 31-Jan-2015].

[Dritsas, 2012] Dritsas, S. (2012). Rationalisation of complex building envelopes. In *Beyond codes and pixels: Proceedings of the 17th International Conference on Computer-Aided Architectural Design Research in Asia*, CAADRIA, pages 16–Jul, Chennai, India. The Association for Computer-Aided Architectural Design Research in Asia, The Association for Computer-Aided Architectural Design Research in Asia.

[Dritsas et al., 2013] Dritsas, S., Kalvo, R., and Sevtsuk, A. (2013). Packing optimization for digital fabrication. In *eCAADe 2013: Computation and Performance–Proceedings of the 31st International Conference on Education and research in Computer Aided Architectural Design in Europe, Delft, The Netherlands, September 18-20, 2013*. Faculty of Architecture, Delft University of Technology; eCAADe (Education and research in Computer Aided Architectural Design in Europe).

[Eigensatz et al., 2010a] Eigensatz, M., Deuss, M., Schiftner, A., Kilian, M., Mitra, N. J., Pottmann, H., and Pauly, M. (2010a). Case studies in cost-optimized paneling of architectural freeform surfaces. *Advances in Architectural Geometry 2010*, pages 49–72.

[Eigensatz et al., 2010b] Eigensatz, M., Kilian, M., Schiftner, A., Mitra, N. J., Pottmann, H., and Pauly, M. (2010b). Paneling architectural freeform surfaces. In *ACM Transactions on Graphics (TOG)*, volume 29, page 45. ACM.

[Evins et al., 2012] Evins, R., Joyce, S. C., Pointer, P., Sharma, S., Vaidyanathan, R., and Williams, C. (2012). Multi-objective design optimisation: getting more for less. In *Proceedings of the ICE-Civil Engineering*, volume 165, pages 5–10. Thomas Telford.

[Few, 2006] Few, S. (2006). *Information dashboard design*. O'Reilly.

[Flood, 1993] Flood, R. L. (1993). *Dealing with complexity: an introduction to the theory and application of systems science*. Springer Science & Business Media.

[Flöry et al., 2013] Flöry, S., Nagai, Y., Isvoranu, F., Pottmann, H., and Wallner, J. (2013). Ruled free forms. In Hesselgren, L. et al., editors, *Advances in Architectural Geometry 2012*, pages 57–66. Springer.

[Foster and Partners, 2013] Foster and Partners (2013). Thames hub airport. http://www.fosterandpartners.com/media/1100160/Foster-Partners_Thames_Hub_Report_July_2013.pdf. [Online; accessed 23-Feb-2015].

[Foster, 2008] Foster, N. (2008). *Foster: catalogue 2008*. Prestel Pub.

[Frazer, 1995] Frazer, J. (1995). *An evolutionary architecture*. AA press.

[Gero, 1990] Gero, J. S. (1990). Design prototypes: a knowledge representation schema for design. *AI magazine*, 11(4):26.

[Gero, 1994] Gero, J. S. (1994). Towards a model of exploration in computer-aided design. In *Formal design methods for CAD*, pages 315–336.

[Gero and Kannengiesser, 2004] Gero, J. S. and Kannengiesser, U. (2004). The situated function–behaviour–structure framework. *Design studies*, 25(4):373–391.

[Gero and Kazakov, 1998] Gero, J. S. and Kazakov, V. A. (1998). Evolving design genes in space layout planning problems. *Artificial Intelligence in Engineering*, 12(3):163–176.

[Goldberg, 1989] Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning*. Addison-Wesley, Reading, MA.

[Gould, 2000] Gould, S. J. (2000). *Wonderful life: the Burgess Shale and the nature of history*. Random House.

[Gramazio and Kohler, 2008] Gramazio, F. and Kohler, M. (2008). *Digital materiality in architecture*, volume 1. Lars Müller Publishers Baden.

[Hanna, 2007] Hanna, S. (2007). Defining implicit objective functions for design problems. In *Proceedings of the 9th annual conference on Genetic and evolutionary computation*, pages 2013–2020. ACM.

[Harding, 2015] Harding, J. (2015). *Meta-Parametric Design*. PhD thesis, University of Bath.

[Harding et al., 2013] Harding, J., Joyce, S., Shepherd, P., and Williams, C. (2013). Thinking topologically at early stage parametric design. In *Advances in Architectural Geometry 2012*, pages 67–76. Springer.

[Haykin and Network, 2004] Haykin, S. and Network, N. (2004). A comprehensive foundation. *Neural Networks*, 2(2004).

[Heathrow, 2014] Heathrow (2014). Heathrow flight paths. `http://your.heathrow.com/under-the-flight-path-reducing-noise/`. [Online; accessed 31-Jan-2015].

[Holland, 1975] Holland, J. H. (1975). *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence.* U Michigan Press.

[Holzer, 2011] Holzer, D. (2011). Bim's seven deadly sins. *International Journal of Architectural Computing*, 9(4):463–480.

[Hudson, 2010] Hudson, R. (2010). *Strategies for parametric design in architecture: an application of practice led research*. PhD thesis, University of Bath.

[Jackson, 1991] Jackson, M. C. (1991). *Systems methodology for the management sciences*. Springer Science & Business Media.

[Jo and Gero, 1998] Jo, J. H. and Gero, J. S. (1998). Space layout planning using an evolutionary approach. *Artificial Intelligence in Engineering*, 12(3):149–162.

[Jones, 1992] Jones, J. C. (1992). *Design methods*. John Wiley & Sons.

[Joyce, 2008] Joyce, S. (2008). The use of computer methods to imitate biological trees and natural optimisation. Master's thesis, University of Bath.

[Joyce et al., 2011] Joyce, S., Fisher, A., Sharma, S., and Williams, C. (2011). Towards ubiquitous structural frame design tools. *Proceedings of IABSE-IASS 2011: Taller, Longer, Lighter Symposium*.

[Joyce, 2015] Joyce, S. C. (2015). Web based data visualisation applied to creative decision making in parametric structural design. In *Proceedings of the International Association for Shell and Spatial Structures (IASS), Future Visions, 17 - 20 August 2015, Amsterdam*.

[Kaijima and Michalatos, 2008] Kaijima, S. and Michalatos, P. (2008). Computational design consultancy. In *Architecture in Computro [26th eCAADe Conference Proceedings/ISBN 978-0-9541183-7-2] Antwerpen (Belgium)*, pages 311–318.

[Kaijima and Michalatos, 2014] Kaijima, S. and Michalatos, P. (2014). Millipede. `http://www.sawapan.eu/sections/section88_Millipede/files/MillipedeMarch2014.pdf`. [Online; accessed 23-Feb-2015].

[Kilian, 2006] Kilian, A. (2006). *Design exploration through bidirectional modeling of constraints*. PhD thesis, MIT.

[Kilian and Ochsendorf, 2005] Kilian, A. and Ochsendorf, J. (2005). Particle-spring systems for structural form finding. *Journal-International Association for Shell and Spatial Structures*, 148:77.

[Koza, 1992] Koza, J. R. (1992). *Genetic programming: on the programming of computers by means of natural selection*. MIT press.

[Kron and Slesin, 1978] Kron, J. and Slesin, S. (1978). *High-tech: the industrial style and source book for the home*. C. N. Potter : distributed by Crown Publishers.

[Legendre, 2011] Legendre, G. L. (2011). Ijp explained: Parametric mathematics in practice. *Architectural Design*, 81(4):44–53.

[Liu et al., 2006] Liu, Y., Pottmann, H., Wallner, J., Yang, Y.-L., and Wang, W. (2006). Geometric modeling with conical meshes and developable surfaces. In *ACM Transactions on Graphics (TOG)*, volume 25, pages 681–689. ACM.

[Maleczek et al., 2013] Maleczek, R., Genevaux, C., and Joyce, S. C. (2013). Linear folded v-shaped stripes. In *Proceedings of the Design Modeling Symposium Berlin 2013*, pages 335–346. epubli GmbH, Berlin.

[Malm et al., 2015] Malm, H., Joyce, S., Tsigkari, M., El-Ashry, K., and Aish, F. (2015). Designing the desert. In *Modelling Behaviour*, pages 159–168. Springer International Publishing.

[Mattheck, 1998] Mattheck, C. (1998). *Design in nature: learning from trees*. Springer.

[Matticus, 2006] Matticus, . (2006). Schematic diagram of the progression of the evolution of the eye. `http://en.wikipedia.org/wiki/Evolution_of_the_eye#mediaviewer/File:Diagram_of_eye_evolution.svg`. [Online; accessed 23-Feb-2015].

[McCandless, 2012] McCandless, D. (2012). *Information is beautiful*. Collins.

[McCarthy, 1990] McCarthy, T. J. (1990). Introduction. In *AutoCAD Express*, pages 1–5. Springer.

[Mendez, 2014] Mendez, T. (2014). *Computational Search in Architectural Design*. PhD thesis, Politecnico di Torino.

[Miller and Thomson, 2000] Miller, J. F. and Thomson, P. (2000). Cartesian genetic programming. In *Genetic Programming*, pages 121–132. Springer.

[Mirtschin, 2011] Mirtschin, J. (2011). Engaging generative bim workflows. *Geometry Gym website*.

[Moretti et al., 2002] Moretti, L., Bucci, F., Mulazzani, M., et al. (2002). *Luigi Moretti: Works and Writings*. Princeton Architectural Press.

[Mueller and Ochsendorf, 2011] Mueller, C. and Ochsendorf, J. (2011). An interactive evolutionary framework for structural design. In *7th International Seminar of the IASS Structural Morphology Group, London*.

[Murdock, 2009] Murdock, K. L. (2009). *Google SketchUp and SketchUp Pro 7 Bible*, volume 606. John Wiley & Sons.

[Ohmori, 2011] Ohmori, H. (2011). Computational morphogenesis: Its current state and possibility for the future. *International Journal of Space Structures*, 26(3):269–276.

[Otto, 1978] Otto, F. (1978). *IL 13: Multihalle Mannheim*. Institute for Lightweight Structures, University of Stuttgart.

[Pena De Leon, 2014] Pena De Leon, A. (2014). *Separation of concerns: strategies for complex parametric design modelling*. PhD thesis, RMIT University.

[Peters, 2007] Peters, B. (2007). *The Smithsonian Courtyard Enclosure: a case-study of digital design processes*, pages 74–83. Riverside Architectural Press.

[Peters, 2008] Peters, B. (2008). The copenhagen elephant house: a case study of digital design processes. In *Proceedings of the 28th Annual Conference of the Association for Computer Aided Design in Architecture (ACADIA)*, pages 134–41.

[Peters, 2013] Peters, B. (2013). Computation works: The building of algorithmic thought. *Architectural Design*, 83(2):8–15.

[Peters and De Kestelier, 2006] Peters, B. and De Kestelier, X. (2006). The work of foster and partners specialist modelling group. In *Proceedings of the Bridges Conference: Mathematical Connections in Art, Music and Science*.

[Peters and Peters, 2013] Peters, T. and Peters, B. (2013). *Inside Smartgeometry: Expanding the Architectural Possibilities of Computational Design*. AD Smart. Wiley.

[Piker, 2013] Piker, D. (2013). Kangaroo: form finding with computational physics. *Architectural Design*, 83(2):136–137.

[Piker, 2014] Piker, D. (2014). personal communication.

[Pottmann et al., 2007] Pottmann, H., Asperl, A., Hofer, M., and Kilian, A. (2007). *Architectural Geometry*. Bentley Institute Press.

[Pottmann et al., 2008] Pottmann, H., Schiftner, A., and Wallner, J. (2008). Geometry of architectural freeform structures. In *Symposium on Solid and Physical Modeling*, page 9.

[Price, 2003] Price, C. (2003). *Cedric Price: the square book*. Wiley-Academy.

[Rittel and Webber, 1973] Rittel, H. W. and Webber, M. M. (1973). Dilemmas in a general theory of planning. *Policy sciences*, 4(2):155–169.

[Rolvink et al., 2010] Rolvink, A., Breuder, J., and Coenders, J. (2010). Structural components-a parametric associative design toolbox for conceptual structural design. In *Symposium of the International Association for Shell and Spatial Structures (50th. 2009. Valencia). Evolution and Trends in Design, Analysis and Construction of Shell and Spatial Structures: Proceedings*. Editorial Universitat Politècnica de València.

[Rosling, 2006] Rosling, H. (2006). Ted talk: Hans rosling shows the best stats youve ever seen.

[Russell and Norvig, 1995] Russell, S. and Norvig, P. (1995). *Artificial Intelligence A Modern Approach*. Prentice-Hall.

[Rutten, 2012a] Rutten, D. (2012a). Grashopper3d forum open a gh automatically. `http://www.grasshopper3d.com/forum/topics/open-a-gh-automatically`. [Online; accessed 23-Feb-2015].

[Rutten, 2012b] Rutten, D. (2012b). Grasshopper. *Generative modeling for Rhino*.

[Rutten, 2014] Rutten, D. (2014). Rhino grasshopper 2.0. personal communication.

[Sakai and Tsunoda, 2015] Sakai, Y. and Tsunoda, D. (2015). Implementation of decentralized version control in collective design modelling. In *Modelling Behaviour*, pages 383–395. Springer.

[Sasaki et al., 2007] Sasaki, M., Itō, T., and Isozaki, A. (2007). *Morphogenesis of flux structure*. AA Publications.

[Schiftner, 2007] Schiftner, A. K. (2007). Planar quad meshes from relative principal curvature lines. Master's thesis, TU Groningen.

[Schramm et al., 1999] Schramm, U., Thomas, H., Zhou, M., and Voth, B. (1999). Topology optimization with altair optistruct. In *Proceedings of the Optimization in Industry II Conference. Banff, Canada*.

[Schumacher, 2009] Schumacher, P. (2009). Parametricism: A new global style for architecture and urban design. *Architectural Design*, 79(4):14–23.

[Segaran, 2007] Segaran, T. (2007). *Programming collective intelligence: building smart web 2.0 applications*. O'Reilly Media.

[Senge, 2014] Senge, P. M. (2014). *The fifth discipline fieldbook: Strategies and tools for building a learning organization*. Crown Business.

[Shelden, 2002] Shelden, D. R. (2002). *Digital surface representation and the constructibility of Gehry's architecture*. PhD thesis, Massachusetts Institute of Technology.

[Shepherd et al., 2011] Shepherd, P., Hudson, R., and Hines, D. (2011). Aviva stadium: A parametric success. *International Journal of Architectural Computing*, 9(2):167–186.

[Shrubshall and Fisher, 2011] Shrubshall, C. and Fisher, A. (2011). The practical application of structural optimisation in the design of the louvre abu dhabi. In *Proceedings of the International Association for Shell and Spatial Structures Symposium. London*.

[Smith, 2007] Smith, S. (2007). Engineering sidra trees. `http://www.architectureweek.com/2008/0227/tools_1-1.html`. [Online; accessed 25-Jan-2015].

[Spencer, 1896] Spencer, H. (1896). *The Principles of Biology*, volume 1. D. Appleton.

[Steenson, 2014] Steenson, M. W. (2014). *Architectures of Information: Christopher Alexander, Cedric Price, and Nicholas Negroponte & MITs Architecture Machine Group*. PhD thesis, Princeton University.

[Stine, 2013] Stine, D. J. (2013). *Design Integration Using Autodesk Revit 2014*. SDC Publications.

[Sudjic, 2010] Sudjic, D. (2010). *Norman Foster: a life in architecture*. Hachette UK.

[Sutherland, 1964] Sutherland, I. E. (1964). Sketch pad a man-machine graphical communication system. In *Proceedings of the SHARE design automation workshop*, pages 6–329. ACM.

[Teixidor, 2007] Teixidor, C. (2007). Hotel hesperia's glazed dome, barcelona, spain. *Structural engineering international*, 17(1):53–55.

[Times, 2013] Times, S. (2013). Captain boris alters london airport plans. `http://www.thesundaytimes.co.uk/sto/news/uk_news/National/article1287453.ece`. [Online; accessed 23-Feb-2015].

[Tsigkari et al., 2013] Tsigkari, M., Chronis, A., Joyce, S. C., Davis, A., Feng, S., and Aish, F. (2013). Integrated design in the simulation process. In *Proceedings of the Symposium on Simulation for Architecture & Urban Design*, page 28. Society for Computer Simulation International.

[Tufte, 1997] Tufte, E. R. (1997). *Visual and statistical thinking: Displays of evidence for decision making*. Graphics.

[Tufte and Graves-Morris, 1983] Tufte, E. R. and Graves-Morris, P. (1983). *The visual display of quantitative information*, volume 2. Graphics press Cheshire, CT.

[Turing, 1950] Turing, A. M. (1950). Computing machinery and intelligence. *Mind*, pages 433–460.

[Turing, 1952] Turing, A. M. (1952). The chemical basis of morphogenesis. *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences*, 237(641):37–72.

[UK Govnemnet, 2011] UK Govnemnet, C. O. (2011). Government construction strategy. `https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/61152/Government-Construction-Strategy_0.pdf`. [Online; accessed 02-Jan-2015].

[Valentine et al., 1999] Valentine, J. W., Jablonski, D., and Erwin, D. H. (1999). Fossils, molecules and embryos: new perspectives on the cambrian explosion. *Development*, 126(5):851–859.

[Victor, 2011] Victor, B. (2011). Explorable explanations. *Bret Victor*, 10.

[Vier, 2013] Vier, R. (2013). Octopus a plug-in for applying evolutionary principles to parametric design. `http://www.grasshopper3d.com/group/octopus`. [Online; accessed 23-Feb-2015].

[von Buelow, 2012] von Buelow, P. (2012). Paragen: Performative exploration of generative systems. *Journal of the International Association for Shell and Spatial Structures*, 53(4):271–284.

[Whitehead and Josefsson, 2011] Whitehead, H. and Josefsson, K. (2011). Architect and mathematician: Designing the dialogue. = http://vimeo.com/34121029. [Online; accessed 23-Feb-2015].

[Williams, 2004] Williams, C. (2004). Design by algorithm. In Leach, N., Turnbull, D., and Williams, C., editors, *Digital tectonics*, pages 78–85. Wiley, Chichester, UK.

[Williams et al., 2014] Williams, C., Malek, S., Joyce, S., and Aish, F. (2014). The use of a particle method for the modelling of isotropic membrane stress for the form finding of shell structures. *Computer-Aided Design*.

[Williams, 2001] Williams, C. J. (2001). The analytic and numerical definition of the geometry of the british museum great court roof. In *Proceedings of the the Third International Conference on Mathematics & Design*. Deakin University.

[Williams, 2013] Williams, O. (2013). So much for 'boris island': London mayor now favours giant four-runway airport in kent over thames estuary development. `http://www.dailymail.co.uk/news/article-2362804/So-Boris-Island-London-mayor-favours-giant-runway-airport-Kent-Thame` `html`. [Online; accessed 23-Feb-2015].

[Willmann et al., 2012] Willmann, J., Augugliaro, F., Cadalbert, T., D'Andrea, R., Gramazio, F., and Kohler, M. (2012). Aerial robotic construction towards a new field of architectural research. *International journal of architectural computing*, 10(3):439–460.

[Wolff et al., 1986] Wolff, J., Maquet, P., and Furlong, R. (1986). *The law of bone remodelling*. Springer-Verlag Berlin.

[Woodbury, 2010] Woodbury, R. (2010). *Elements of parametric design*. Taylor and Francis.

[Woodbury et al., 2007] Woodbury, R., Aish, R., and Kilian, A. (2007). Some patterns for parametric modeling. In *27th Annual Conference of the Association for Computer Aided Design in Architecture*, pages 222–229.

[Xie and Steven, 1993] Xie, Y. and Steven, G. P. (1993). A simple evolutionary procedure for structural optimization. *Computers & structures*, 49(5):885–896.

[Zadravec et al., 2010] Zadravec, M., Schiftner, A., and Wallner, J. (2010). Designing quad-dominant meshes with planar faces. *Computer Graphics Forum*, 29(5):1671–1679.

[Zhang et al., 2008] Zhang, G.-Q., Zhang, G.-Q., Yang, Q.-F., Cheng, S.-Q., and Zhou, T. (2008). Evolution of the internet and its cores. *New Journal of Physics*, 10(12):123027.

Appendix A

Selected Published Papers

Below are the published papers developed by the author and other contributors over the duration of the EngD research and referenced in this work. They are in order sorted by date as below:

Malm, H., Joyce, S., Tsigkari, M., El-Ashry, K., and Aish, F. (2015). Designing the desert. In *Modelling Behaviour*, pages 159–168. Springer International Publishing

Joyce, S. C. (2015). Web based data visualisation applied to creative decision making in parametric structural design. In *Proceedings of the International Association for Shell and Spatial Structures (IASS), Future Visions, 17 - 20 August 2015, Amsterdam*

Tsigkari, M., Chronis, A., Joyce, S. C., Davis, A., Feng, S., and Aish, F. (2013). Integrated design in the simulation process. In *Proceedings of the Symposium on Simulation for Architecture & Urban Design*, page 28. Society for Computer Simulation International

Williams, C., Malek, S., Joyce, S., and Aish, F. (2014). The use of a particle method for the modelling of isotropic membrane stress for the form finding of

shell structures. *Computer-Aided Design*

Harding, J., Joyce, S., Shepherd, P., and Williams, C. (2013). Thinking topologically at early stage parametric design. In *Advances in Architectural Geometry 2012*, pages 67–76. Springer

Maleczek, R., Genevaux, C., and Joyce, S. C. (2013). Linear folded v-shaped stripes. In *Proceedings of the Design Modeling Symposium Berlin 2013*, pages 335–346. epubli GmbH, Berlin

Aish, R., Joyce, S., Fisher, A., and Marsh, A. (2012). Progress towards multi-criteria design optimisation using designscript with smart form, robot structural analysis and ecotect building performance analysis. In *Synthetic Digital Ecologies, ACADIA Conference Proceedings, San Francisco, USA, 18–21/10/12, pp 47-56*, page 10. ACADIA

Evins, R., Joyce, S. C., Pointer, P., Sharma, S., Vaidyanathan, R., and Williams, C. (2012). Multi-objective design optimisation: getting more for less. In *Proceedings of the ICE-Civil Engineering*, volume 165, pages 5–10. Thomas Telford

Joyce, S., Fisher, A., Sharma, S., and Williams, C. (2011). Towards ubiquitous structural frame design tools. *Proceedings of IABSE-IASS 2011: Taller, Longer, Lighter Symposium*

Designing the Desert

Modelling the Walls of the UAE Pavilion for the 2015 World Expo

Henrik Malm, Sam Joyce, Martha Tsigkari, Khaled El-Ashry and Francis Aish

Abstract The design and modelling of the walls of the United Arab Emirates Pavilion for the 2015 World Expo in Milan was a rich and complex challenge in shape research, computational design, digital fabrication and considerations for logistics and construction. This paper will focus on how the shape and textures of the walls of the pavilion were developed and specifically presents a novel method for generating tileable, three-dimensional, sand ripple patterns based on the theory of reaction-diffusion equations. The walls were to be panelised by glass fibre reinforced concrete panels and the paper also explains how a multi-objective optimisation approach was applied to maximise the randomness of the placement of the different textured panels under budget and fabrication constraints.

1 Introduction

The national pavilions at the World Exhibitions have often tried to express certain recognizable flavours of the specific countries and the United Arab Emirates pavilion for the 2015 World Expo in Milan follows in this tradition. Under this spectrum, the design brief was to create walls inspired by the stunning desert landscapes of the United Arab Emirates. The walls were to be panelised using Glass fibre Reinforced Concrete panels (GRC). The current paper will tell the story of the design and modelling of these walls and in the next section, Section 2, we will describe the research conducted on the shape of sand dunes and how the walls found their main form based on this. Then, in Section 3, we will zoom in on the computational generation of the smaller scale ripple pattern that covers a large part of the panels. It

Henrik Malm, Sam Joyce*, Martha Tsigkari, Khaled El-Ashry and Francis Aish
Foster + Partners, 22 Hester Road, London SW11 4AN, UK
e-mail: {hmalm|mtsigkar|kelashry|faish}@fosterandpartners.com

* Currently with Singapore University of Technology and Design
e-mail: {sam_joyce}@sutd.edu.sg

will be explained how this pattern was generated in order to produce a tileable result and we will cover both how the 2D topology of the pattern was found and how this then was used to create the final 3D texture on the panels. Further, in Section 4, a multi-objective optimisation approach will be outlined, which was used to find the optimal sequence of the panels along the walls, maximizing the randomness of the panels while minimising the total number of moulds. The paper is rounded off in Section 5 and 6 with our acknowledgements and conclusions.

2 The Shapes and Textures of the Desert

To find the form and textures of the pavilion walls, research on the shapes created by the natural elements in the sandy landscapes of the UAE deserts was conducted. In this process, it was early on observed that features on two main scales dominate the typical perception of sand dunes; At a large scale, the wave shaped dune formation dominates the picture, while at a smaller scale an elongated fine-grained sand ripple pattern is evident, Fig. 1. The physics of the formation of these dune and ripple structures has been extensively studied and simulated by other authors, c.f. (Nishimori and Ouchi, 1993; Yizaq et al., 1991; Lamb et al., 2002; Tsoar, 2001). Briefly, the small-scale ripple pattern is spontaneously formed by the saltation of sand grains when the wind force exceeds a critical value, while the large scale dune is gradually built up by the prevailing wind until it collapses on the leeward side.

In the current project, field trips were made in order to study and register real sand dune and ripple formations. The shape of several examples of sand ripples were 3D scanned and the resulting models 3D colour printed. This gave us the possibility to study the shape and proportions of the ripples, together with a true material and shadow perception, Fig. 2. Measurements on site in the desert outside Abu Dhabi were also made in order to get a first-hand feeling for the proportions, Fig. 3.

It was concluded that the features at the two main scales of the desert formations, the dunes and the ripples, should be expressed together in an important part of the building; the entrance. Following elongated site constraints, this part was designed

Fig. 1: The two scales of sand dunes.

Fig. 2: 3D prints of sand ripples.

Fig. 3: Measurements on site outside Abu Dhabi.

as a canyon, with narrow passages followed by more open areas. Here the visitors should be totally immersed in the experience of the dunes. The inner walls were designed with an undulating ridge on each side of the canyon reaching through the total length of the structure. On the upper part of these ridge walls, portraying the windward side of the dune, the sand ripple pattern was applied, while a smooth surface was modelled on the lower part, portraying the leeward side. See Fig. 4 for a visualization of the final canyon geometry at the entrance location and Fig. 6 for a plan view of the canyon.

There was also a wish to create some shelter from the sun or rain within the canyon and an array of different deployable structures for this were considered. However, in the end the undulating ridge of the walls was used for the purpose by allowing it to protrude further out into the canyon at the more open areas, in order for the structure itself to create an amount of self-shading and shelter at those locations. In the narrower areas the ridges were instead modelled close to the baseline of the walls, creating a dynamic entrance sequence, c.f. Fig. 4, 5 and 6.

Fig. 4: Render of the entrance to the canyon.

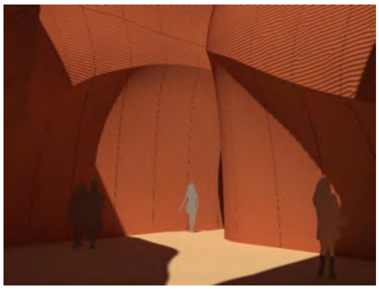


Fig. 5: The self-shading of the canyon.

Fig. 6: Plan view of the entrance canyon. Fig. 4 and 5 are rendered at the marked locations.

3 Computational Generation of Ripples

3.1 Topology

In contrast to the canyon, the ripple pattern covers the whole 12m height of the panels on the rest of the GRC walls. In order for these GRC panels to have some repetition, and to not all require unique moulds, a novel method for constructing a tileable ripple pattern was developed within the project. The repetition of these panels was also made possible by the fact that they are all straight extrusions from the base-line curves of the walls, c.f. Section 4. In the initial design, these panels covered most of the exterior and interior walls, but in a later value-engineering stage only the exterior walls were clad with GRC. All of the panels of the canyon walls are still bespoke because of their free-form doubly curved shapes.

Some different approaches to 2D pattern generation were reviewed early in the process, such as work by Turner (Turner), but it was quite soon observed that the solution space of a basic reaction-diffusion equation bore similarity to the branching structures of sand ripple formations. The theory of reaction-diffusion equations and their shape generative properties was first studied by Turing (Turing, 1952), while more recently other authors have studied the texture and shape generating properties of these equations within the field of Computer Graphics, c.f. (Witkin, 1991; Sanderson et al., 2006). In order to produce the elongated branching topology of the sand ripple pattern, an anisotropic reaction-diffusion equation was needed where the diffusion coefficient is replaced by a diffusion function which depends on the spatial directions (i.e. the amount of diffusion for "the chemical species" can be different in the x- and y-directions). To this end, we started looking at a solver implemented by Nervous System (Inc) in the Java-based coding environment Processing (Fry and Reas), which we then modified as described below in order to meet our specific needs. Another implementation of a reaction-diffusion solver considered early in the process was the work by Schmidt in his toxiclibs libraries for Processing (Schmidt).

The anisotropic reaction-diffusion equations used in this project follows the model of Gray-Scott,

$$\frac{\partial u}{\partial t} = r_u(\nabla u)\nabla^2 u + uv^2 + f(1-u) \quad (1)$$

$$\frac{\partial v}{\partial t} = r_v(\nabla v)\nabla^2 v - uv^2 - (f+k)v \quad (2)$$

, where u and v represent the concentrations of two chemical species U and V. $r_u(u)$ and $r_v(v)$ are their diffusion rates, which in this anisotropic case depends on the direction of the gradients, ∇u and ∇v, of the concentrations in the current point. In the full explanation of the Gray-Scott model a third species P is also introduced and parameter k can then be interpreted as the rate of conversion of the species V to P and f as the amount of feeding of the species U and the draining of the species V and P.

By randomizing the initial conditions (seed) for the solver, new patterns were created for every simulation, giving an infinite range of branching patterns. As for all diffusion equations, this anisotropic reaction-diffusion equation also comes close to an equilibrium state for higher time values t, which with our settings meant a pattern of totally parallel lines (the diffusion of the dominating species is amplified in the x-direction, while the reaction and diffusion is set more equal in the y-direction). In order to instead get the desired amount of branching bifurcations in the pattern, the simulations needed to be supervised and the solver stopped at the right time giving the desired density of bifurcations.

However, for different seeds the above process outputs individual unrelated patterns that do not match when placed next to each other. The main challenge was now to create a process where the boundary conditions of the solver could be controlled in order to give a pattern that was continuous from tile to tile, both in position and in the tangents of the central curves of the ripple ridges. To this end, the first step in the process was to generate an initial base pattern. The boundary of this first pattern was stored in order to then be used as a part of the initial conditions of the rest of the ripple tile simulations, making all the generated patterns match. However, it was not enough to only store the very edge of the first generated pattern. In order to get the tangents of the ripples to match, which of course is very important for the flow of the pattern across panels, a wider edge band from the initial simulation needed to be stored. What was stored from the initial simulation was actually not the generated grey level of the pattern image, but the gradients of the intensity levels (i.e. the gradients of the relative amount of the chemical substances simulated, in the original interpretation of the equation). In the final simulation process, by weighting the gradients from the border conditions with a standard unbiased simulation, i.e. without border conditions, a satisfying flow of the ripples could be achieved, without any kinks or discontinuities.

This process gave us the possibility to simulate a theoretically infinite number of matching ripple textures. The process could be used to generate patterns matching on all four edges of a rectangular tile, however, in this project we finally chose to generate full height panels in one go, so that the patterns only needed to match along the horizontal direction. See Fig. 7 for a set of matching pattern tiles (cropped in height).

Fig. 7: An example of pattern tiles generated by the tileable reaction diffusion solver.

3.2 3D Structure

The process outlined above only gives the basic topology of the pattern. In order to actually achieve a 3D ripple pattern, with its wave like cross section, additional processing was required. To this end, the essential central line of the ripple curves was extracted. This was realised by initially thresholding the output images from the reaction-diffusion solver into binary black and white images. A process called skeletonization, c.f. (Gonzalez and Woods, 1992), was then applied, resulting in only the central line of the pattern stripes being preserved. This method was implemented using Python and its OpenCV libraries (Itseez). See Fig. 8, for a set of matching skeletonized patterns.

The next step was to replace these central line curves with a 3D geometry giving the correct cross section of the ripples. By studying our scanned ripple specimens and our on-site measurements, a number of options for the final cross section were designed, see Fig. 9 for a selection. The first method developed for generating the 3D geometry was based on 2D image filtering, where all the white ”on” pixels in the binary skeleton images where replaced by a special filtering kernel. Interpreting the intensity of this kernel as a height measure, the kernel was created so that it had the correct 3D ripple section. Its pixel values were additionally chosen with a Gaussian fall-off to the outer edge of the kernel, so that when applying the kernel

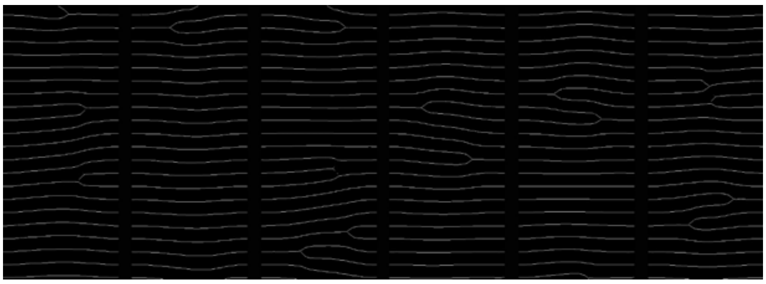


Fig. 8: Skeletonization of matching tiles of the reaction-diffusion pattern.

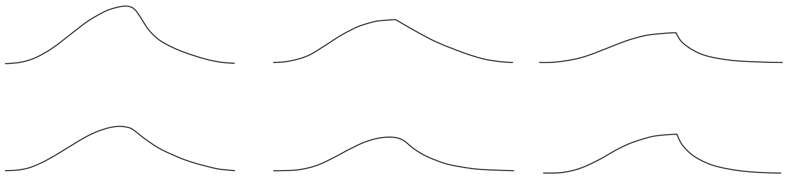


Fig. 9: Example of ripple sections. An option close to the lower right section was finally chosen.

the resulting output image had a smooth transition in every direction, except in the direction orthogonal to the ripple ridge. The resulting image was then used as 3D displacement data in order to obtain the actual 3D geometry. However, this approach had two major drawbacks. Firstly, despite the intrinsic smoothing in the above mentioned process, the image was still pixelated and gave a quite jittery look of each ripple ridge. Secondly, the resulting geometry became extremely heavy and would have been very difficult to work with in later stages.

To overcome the problems of the previous method, a different approach was subsequently developed where the pixelated curves in the skeleton images were divided into a number of control points. These control points were then interpolated into a set of NURBS curves, which was in turn used as rail curves along which a curve with the wanted ripple section was swept. This process was not as straightforward as it might seem, since some tricky special cases occurred, especially at the splitting of a rail curve (at the so-called bifurcations). However, the process could in the end be totally automated and resulted in a 3D ripple structure that is very reminiscent of the inspiration. See Fig. 10 for a comparison between a real sand ripple and an example of a computer generated bifurcation on one of the fabricated panels. The bespoke sweeping algorithm was implemented in the parametric modelling plug-in Grasshopper for Rhino 3D (McNeel).

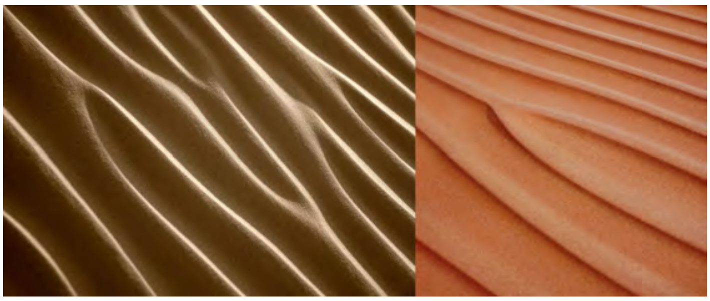


Fig. 10: Comparison of a real desert sand ripple and a detail from one of the final GRC panels.

4 Optimizing the Ripples

Since all panels on the building, excluding the canyon, are straight extrusions from singly curved base lines there was an opportunity for rationalising the geometry of these panels. A ”semi-facetted” approach was chosen for this where the panels were divided into a family of arcs that fit the base lines as closely as possible. In the end, a family of 9 different arcs was chosen, where the radii was found through optimisation with the genetic solver Galapagos in Grasshopper. Additionally, 20 special panels were added, dealing with outliers with too large deviations from the baseline curves. We will not detail the geometry rationalization further here, but instead focus on the succeeding multi-objective optimization that relates to the randomisation of the ripple pattern.

Due to the GRC fabrication process, every combination of ripple pattern and panel arc radius demanded a unique mould to be made. The ripple generation technique outlined in Section 3 above could theoretically produces an infinite number of matching patterns. However, there would not be any benefit of the tiling if the same pattern was not reused at several locations. After visual testing, using rendered images, it was decided that 8 different patterns were sufficient to give a good overall feeling of randomness in the wall texture.

Let us say that all unique arcs in the panel family were only used 8 times or less on the building. Using as many different patterns (of the 8 available) as possible for each arc would then still demand a unique mould for every panel on the building. But, since each mould actually could be reused up to 10 times in the fabrication, there was a wish to get closer to this amount of the usages per mould, and minimize the number of total moulds to be milled and prepared.

To find a solution to the conflicting objectives of reducing the number of moulds while maximizing the randomness of the overall texture, a multi-objective optimisation approach was applied. The solver used was the Octopus plug-in for Grasshopper which implements SPEA2 (Strength Pareto Evolutionary Algorithm) by Zitler et.al. (Zitzler and Thiele, 1991). Two objective functions, f_1 and f_2, were used, where f_1 was based on an adjacency measure $a_{i,j}$ equivalent to the distance from panel i to another panel j along the wall, disregarding arc type. The inverse of $a_{i,j}$ was then summed over the 10 closest panels to panel i, counting only identically patterned panels. This was then summed over all N panels. Let P_i denote the pattern at panel i, then

$$f_1 = \sum_{i=0}^{N} \sum_{j=i-5}^{i+5} b_{i,j} \qquad \text{,where} \qquad b_{i,j} = \begin{cases} 0, & i = j \\ 0, & P_i \neq P_j \\ \frac{1}{a_{i,j}}, & P_i = P_j \end{cases} \tag{3}$$

The second objective function was just the total number of moulds. Since each mould could be reused a maximum of 10 times, this function can be written as

$$f_2 = \sum_{i=0}^{n_A} \sum_{j=0}^{n_P} \lceil \frac{N_{i,j}}{10} \rceil \tag{4}$$

, where $N_{i,j}$ is the number of panels with arc radius i and pattern j, n_A is the total number of arcs and n_P is the total number of patterns.

The genetic algortihm ran through 1018 generation, with a population size of 100. Figure 11 shows the Pareto front and evaluated panel configurations plotted in the objective space after the process had converged, with f_1 on the x-axis and f_2 on the y-axis. Three different non-dominated points on the Pareto front have been selected and Figure 12 shows the corresponding panel configurations, in the same order, from left to right. The panels have been color-coded according to the level of adjacency: Red, orange, yellow, green and grey represents panels with a distance of zero, one, two, three and larger than three panels to a panel with the same ripple pattern. A point on the Pareto front, regarded as the best compromise between total number of moulds and visual appearance, was finally chosen as the final panel configuration. In the end, the multi-objective optimisation process approximately halved the number of moulds needed to be prepared for the GRC fabrication.

Fig. 11: The Pareto front of the multi-objective optimization after 1018 generations. Dark red marks points on the final Pareto front, light red marks previously non-dominated points and light green shows dominated evaluated points.

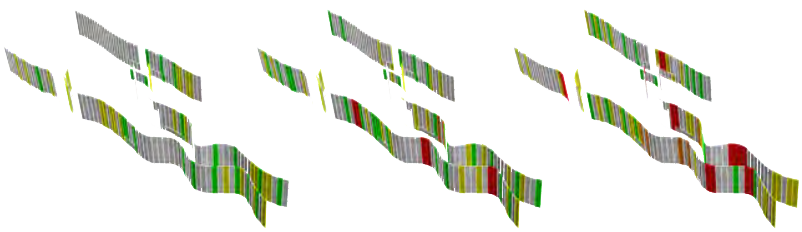


Fig. 12: Panel configurations corresponding to the points on the Pareto front marked in the graph above. Red, orange, yellow, green and grey represents panels with a distance of zero, one, two, three and larger than three panels to a panel with the same ripple pattern. The number of moulds for each configuration is 64, 60 and 55, respectively.

5 Acknowledgements

The authors would like to thank our colleagues and collaborators in the wider project team.

6 Conclusions

The design and modelling of the walls of UAE Expo Pavilion was a complex and inspiring study into natures shape formation processes and the usage of these studies to create an attractive space and building. The walls and entrance space was designed to reflect the perception of the UAE desert landscapes, with the undulating sand dune ridge of the entrance canyon creating a dynamic and self-shading structure. The creation of a tileable 3D sand ripple pattern, based on the theory of reaction-diffusion equations, was one of the major developments within this project, a process which has been a focus of this paper. This pattern was applied on all of the GRC panels on the building. To find a solution to the contrasting objectives of minimizing the number of moulds for the GRC fabrication while preserving a random appearance of the ripple pattern on the panelised walls, a multi-objective optimization strategy was applied, a process which has also been detailed here.

In addition to the subjects covered in this paper, the project also presented challenges when it came to rationalization of the wall geometries and the creation of an efficient pipeline from design to the GRC fabrication. Many aspects concerning the installation and transportation of the panels also had to be considered in the design, setting limits for example of the height and width of each individual panel.

The pavilion was constructed during the autumn of 2014 and the spring of 2015 and opened on the 1st May 2015. See Fig 13, for some images from the construction phase and the finalized project.

References

B. Fry and C. Reas. Homepage of processing. `https://www.processing.org`. Accessed 15 April 2015.

R. C. Gonzalez and R. E. Woods. *Digital Image Processing*. Prentice Hall, Boston, 2 edition, 1992.

N. S. Inc. Homepage of nervous system. `https://www.n-e-r-v-o-u-s.com`. Accessed 30 April 2015.

Itseez. Homepage of opencv. `https://www.opencv.org`. Accessed 13 April 2015.

P. Lamb, J. Kwan, and S. Ahn. A computer simulation of sand ripple formation. Technical report, Harvey Mudd College Dept of Mathematics, 2002.

R. McNeel. Rhinoceros. `https://www.rhino3D.com`. Accessed 15 April 2015.

H. Nishimori and N. Ouchi. Formation of ripple patterns and dunes by wind-blown sand. *Phys Rev Lett*, 71(1):197, 1993.

A. R. Sanderson, R. M. Kirby, C. R. Johnson, and L. Yang. Advanced reaction-diffusion models for texture analysis. *J Graphics, GPU and Game Tools*, 11(3): 47–71, 2006.

K. Schmidt. Toxiclibs, simutils-0001, gray-scott reaction diffusion. `http://www.openprocessing.org/sketch/2698`. Accessed 01 May 2015.

H. Tsoar. Types of aeolian sand dunes and their formation. *Geomorphological Fluid Mechanics, Lecture Notes in Physics*, 582:403–429, 2001.

A. M. Turing. The chemical basis of morphogenesis. *Phil Trans Roy Soc*, 237: 37–72, 1952.

A. Turner. Sand ripples. processing sketch from open processing. `http://www.openprocessing.org/sketch/2698`. Accessed 01 May 2015.

A. Witkin. Reaction-diffusion textures. *ACM SIGGRAPH Computer Graphics*, 25 (4):299–308, 1991.

H. Yizaq, N. Balmforth, and A. Provenzale. Blown by wind: Non-linear dynamics of aeolian sand ripples. *Physica D: Nonlinear Phenomena*, 195:3–4, 1991.

E. Zitzler and M. L. L. Thiele. Spea2: Improving the strength pareto evolutionary algorithm. Technical Report 103, Dept. Electical Eng., ETH, Zrich, 1991.

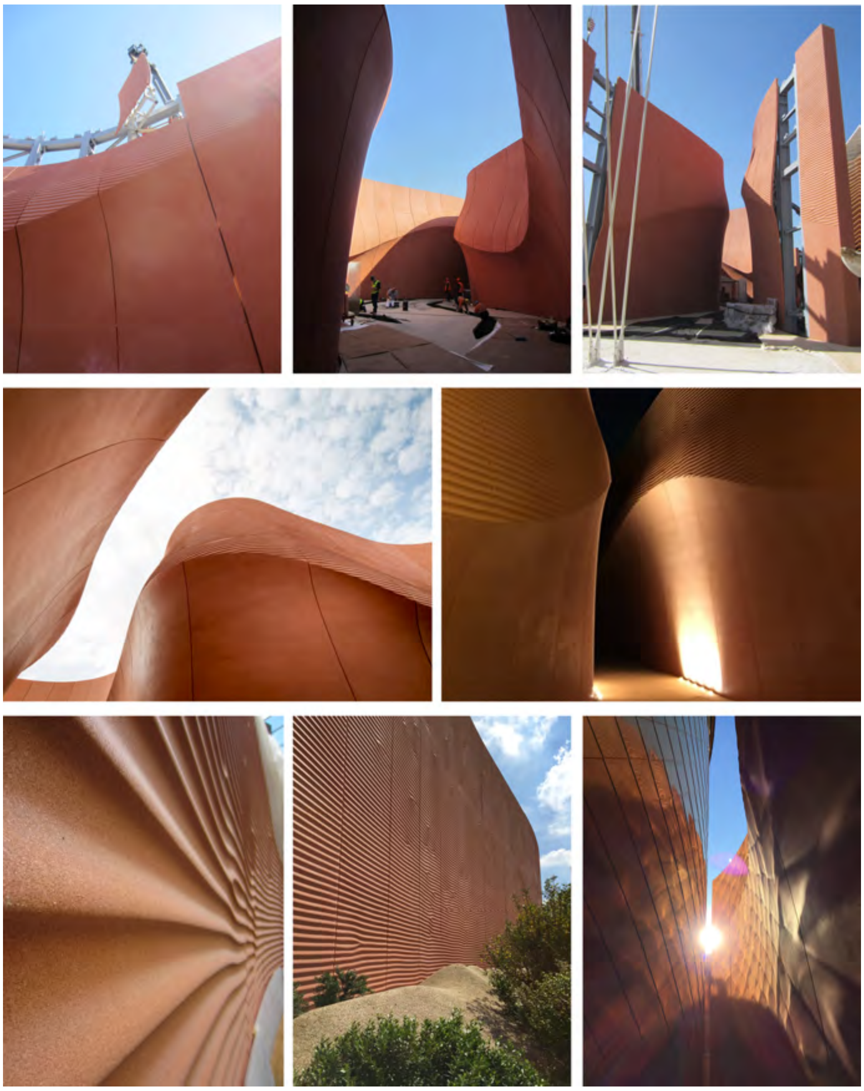


Fig. 13: Photos from the construction phase and the finished project.

Web Based Data Visualisation Applied to Creative Decision Making in Parametric Structural Design

Sam Conrad JOYCE*

*Singapore University of Technology and Design
8 Somapah Road, Singapore, 487372
sam_joyce@sutd.edu.sg

Abstract

This paper outlines recent efforts by the author to integrate modern web-based data visualisation techniques into decision-making efforts for structural design projects. It traces the development of data visualisation, detailing powerful new techniques, which exist in modern web browsers, and are already used by many other data-rich professions. Explaining the underlying technology as compared to current engineering visualisation equivalents. With the benefits and potential applications of this technology to design engineering discussed. Then using real design problems, some example applications of these methods applied to structural engineering decision making support are described and explained for supporting large scale option based decision making with structural engineering data.

Keywords: data visualisation, big data, design decision-making, web technologies, parametric design

1. Introduction

Good engineering is not only about deductively doing the right calculations and finding safe solutions. It is the act of inductively navigating though the wide range of potential options, making informed decisions at every stage. This creative decision making process has become more important as buildings and engineering projects have increased in size and complexity; where early decisions often based on little more than a few studies and intuition, have a large impact on the later configuration and thus performance of a structure.

With the advent of computing, data has been a cheap commodity in engineering analysis, making design studies easier, and more information available to the engineer. However, arguably this has both helped and hindered decision-making. Knowing more, does not necessarily translate into being able to make better decisions; as there is an inductive synthesis and interpretation process also known as understanding, which required before one can begin making decisions using data.

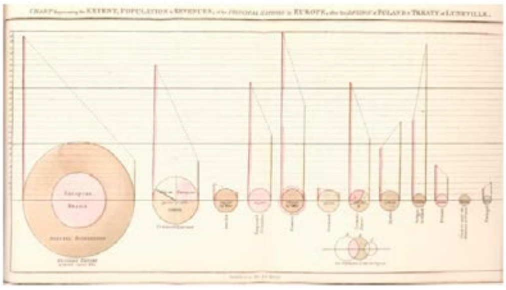


Figure 1: The first known use of the pie chart by William Playfair from his 'Commercial and Political Atlas' published in 1786

In other fields, new techniques have arisen that aim to improve data understanding via better visualisation. Either by increasing the quantity or quality of what is shown, or by taking advantage of the interactivity of screen based visualisations. It is the application of these mediums and techniques in an engineering context that will be explored in this paper.

2. Data Visualisation

Data visualisation is a relatively new field, which at a functional level pairs graphic design with quantitative information, but at a discipline level, studies peoples cognitive understanding and interpretation of graphical figures. With the aim that data can be conveyed in the most efficient, but accurate and representative way. Data visualisation has been practised as early as 1765 by Joseph Priestley [8] with the innovation of timeline charts. The art gained popular appreciation in 1786 with William Playfair in his widely published 'Commercial and Political Atlas' [7], and was being used to steer government policy making by 1855 by John Snow used to prevent further cholera outbreaks [4]. With many of the graphical techniques in use today pioneered by Civil Engineer Charles Joseph Minard in the 1800's [6].

However, arguably the field of data visualisation gained it's academic and scientific foundations after the publishing of 'The Visual Display of Quantitative information' by Edward Tufte [9]. In this work, the principals of clear and concise figures was developed, with the defined aim to help not hinder understanding by the reader, which he coupled with careful analysis on actually how figures are understood by people and if they correlate with the data and intended understanding. These concepts have now become core to the discipline. An important contribution by Jacques Bertin [2], determines 5 principal properties of graphical objects which can be determined by eye namely size, value, texture, colour, orientation and shape and these have differing ability to infer association, selection, order and quantity of data. This effectively defines the base axioms from which modern visualisations can be made. Arguably Tufte's later work 'Visual and Statistical Thinking: Displays of Evidence for Making Decisions' [10], is perhaps most important as it links how visualisations are practically used to make decisions, and how evidence if incorrectly presented can result in the wrong decision being made.

Nowadays the use of data visualisation is ubiquitous, from Power Point figures, to sophisticated graphing methods developed by the financial industry. Furthermore, these methods are gaining more

Figure 2: A spatial data analysis of deaths by Cholera, leading to the discovery of the root of the disease and a decision to stop using a pump which saved many lives.

interest and relevance as the internet and big data are increasing the needs for professionals to digest and act on complex ever changing data sets. There has also been resurgent popular interest in data visualisations with popular publications such as 'Information is Beautiful' by Mc Candless [5]. A growing trend is a move away from static representations, towards more dynamic web based visualisations, which utilise the unique capabilities of the internet 'medium' of computer screen and user interaction, to provide much richer systems, and evolve methods of visual understanding. These are becoming more easily accessible with the new HTML5 and JavaScript standards implemented on modern browsers. Which enable powerful computation, figure generation and animations to be computed on the 'client' (local user's web browser) side.

3. Current State of the Art in Engineering

The visualisation landscape of engineering data used in structural design is still very disjointed and fragmentary. Driven mostly by commercial software, which typically acts monolithically as the platform to build structural models, analyse them and then show the complex data. This is a very demanding set of objectives, and it is felt by the author that often one or two of these capabilities are well realised in a program, but often at the price of others. Looking at the visualisation component of this; Some programs take a very three dimensional model based approach, integrating analysis data with the 3D model, examples such as GSA by Oasys or Robot by Autodesk. Others provide more fixed figure generation methods, such as Ansys or the ETABS software by Computers for Structures.

However almost universally, direct quantitative data is shown in table format, with Excel interoperability a heavy influence on the display of this information. This includes higher-level information such as minim and maxim stresses, average element utilisation etc. This fits into current workflows where often engineers manually copy large tabular information into Excel spreadsheets, which are developed individually by the engineer to generate new tables and further calculations to summarise and digest this information.

It is the belief of the author that whilst this is effective for basic analysis of single designs and code checking, it is less than ideal for exploring different options or making design decisions based on comparative performance data. As engineering analysis is becoming easier and quicker to generate, more analysis is undertaken on a range of options or range of load cases. This is acutely the case with

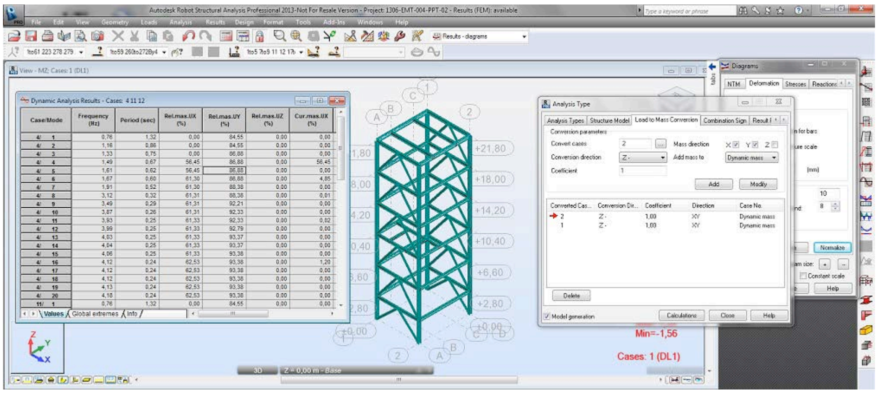


Figure 3: Typical structural analysis software interface from Robot, with 3D model viewer, but results displayed only in a tabulated manor. From http://forums.autodesk.com/

parametric design, here modern systems have enabled geometric modelling to be linked quickly to engineering analysis with results data able to be extracted and processed more fluidly than before as shown by Aish [1]. As such the development and generation of a larger range of options is not only possible, but is becoming to be expected by design teams to hone design variables. It is here, that the use of static plots, of the kind that can be created in Excel, reach their limits of expression, and alternatives need to be sought.

What will be discussed in the rest of this paper are web-based alternatives to exploring complex engineering datasets, with a goal that this should enable wider exploration of design space, whilst supporting informed conclusive design decisions.

4. A Proposal for Modern Data Presentation

The data visualisations proposed in this work are single page web pages. To cope with the complexity modern web development is broken down into three main components, and due to it's use paradigm and historical reasons each has it's own encoding schema. Firstly, is the HTML defining the pages overall structure and content, using a hierarchical 'tree' of 'elements' which are the basic building blocks of a web page, determining where text would be and what it is, as well as where buttons or user input fields are placed. The second component is CSS, which defines the style of the page, such as font types and element colours, and is often referenced across multiple pages to give them a unified aesthetic. Finally there is JavaScript which is compiled and run by the client computer upon loading the web page, and can then provide actions and logic for creating elements, or handling message passing and logic with can be controlled by elements on the web page such as buttons. JavaSript represents a deceptively powerful programming language, which drives many of the features associated with the modern web. Despite this complexity all of these elements can be contained in one page and opened locally from the web browser, in the same manner as any word document but with all the interactively of a webpage.

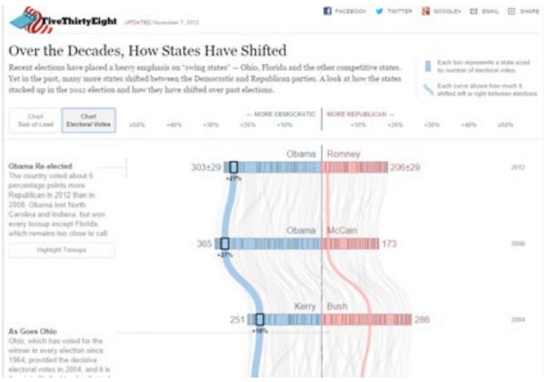


Figure 4: Example of New York Times web page, in which D3 is used to power the visualisations.

4.1. Data visualisation web frameworks

One popular library written in JavaScript specifically for data visualisation is Data-Driven-Documents or 'D3' Bostock et al [3]. This technology has been widely adopted for visualisation on the web including the New York Times website to improve understanding of voting statistics. D3 uses a paradigm where data is bound/linked to web objects, exposing methods for the objects to use the associated data when they are interacted with.

This data is object based, so one element doesn't have to relate to one piece of data, but rather can relate to a complex set of data, allowing for hierarchically nested properties; For example, a set of web elements could be linked to a set of beams, each element would then be able to access it's respective beam's maximum stress (a direct property) or the stresses along a number of points of the beam (properties of a property / collection). The power of this lies in the fact that a graphical web objects such as dots or bars, can be defined or controlled by referring to their inherent data properties, which are bound to the element itself. Furthermore D3 also allows for the creation, destruction and transformation of web objects based on data. For example, making the same number of points, as there are number of nodes in a data list. The position of these point elements in the web page could be linked to the X and Y location of it's linked node data.

4.2. Applications in Engineering Design

If we look at the kind of post-processing of analysis, undertaken by structural engineers, it typically involves assuring the validity of analysis result, such as checking loads and reactions sum to zero. However, when it comes to comparatively appraising actual designs by aggregating the data a sense of distribution of utilisation with respect to design parameters for example can be very useful.

Frameworks such as D3 can provide a level of interactive data display not available in other tools. For example, points that represents nodal data, and when selected based on analysis data (such as high deflection) can then highlight where they exist in physical space. The logic can be built and

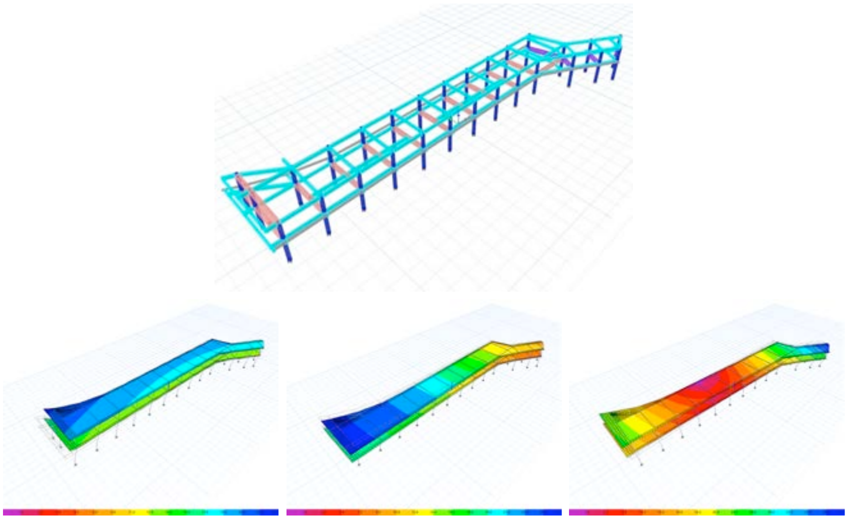


Figure 5: Portal frame considered in study.
Below the first three modes of the portal frame coloured by displacement.

determined by the user, so if indicating where neighbour node are on a graph is useful, the logic can be programmed were chosen nodes then highlight their connected nodes. Furthermore, the input data can be in the form of a basic tabulated text file, which could match an existing FEM software output, and can be easily replaced for new data without needing to modify the web page.

4.3. Support Option Selection

These concepts become more powerful when comparing different options or load cases. In this scenario each option represents a wide range of multifaceted data about that specific design. This causes an issue with static plots, as displaying all of this data requires many different plots. The issue becomes compounded when comparing many potential design options on mass. Furthermore, one representation method for one property of the design may not be the appropriate view for another. For example the maximum deflection is arguably best displayed as a point as it is a singular value, where as the material volume is best represented and compared as a bar chart to give a sense of comparison. Again, web visualisation and specifically D3 can support these requirements by containing each option on the same screen and transitioning between the various data views on command from the user. This centralised but interactive approach can simplify views of data whilst still allowing for depth, and by transitioning the user can get a better sense of persistence over separate plots.

5. Case Studies

This technology is perhaps best explained in the context of example case studies. Presented below are two case studies, both of which are the product of attempting to finding better visualisation solutions to design problems encountered by the author. The first highlights the applicability of such a method

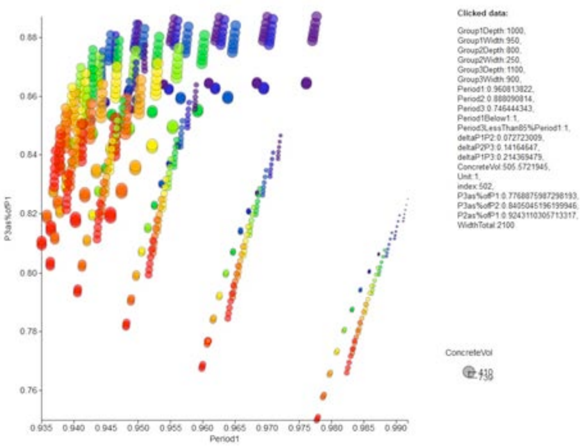


Figure 6: Example of data visualisation, showing trade off between first mode period and difference between first and third mode periods. Point circle area represents material volume, and are coloured by section size. In depth details of a chosen (clicked) point are displayed to the right. The axis properties can be changed to any performance metric by the user.

for understanding engineering trade-off. The other shows how multi-directional engineering and design constraints can be used to improve decision-making.

5.1 Modal Design

In this example a structure with complex dynamic response was investigated. The structure was a rectangular portal frame with asymmetrical additions. With fixed column sizes and beams comprising of three family types where the width but not the depth could be varied. In dynamic analysis the first three modes represented a translation in X, translation in Y and rotation about the Z-axis respectively. The X-axis being the long axis of the structure as shown in figure 5.

The issue was two fold, firstly the structure having a too low period for the first mode, secondly keeping the third mode away from the first mode period, which would combine destructively if not sufficiently separated. The result being a complex iteration of three different modes, controlled by the three different families of beam sections to be designed for the portal frame.

Rather than tune the structure manually which would be a time intensive process, instead an exhaustive parametric analysis was undertaken. Sampling 9 different options for each of the beam section families resulting in 729 potential solutions. After filtering out those, which do not satisfy strict criteria such as minimum deflection, the more subtle process of choosing which is the best trade-off can be undertaken.

This was visualised by graphing these on a two axis scatter plot, where each point represents a design. Two extra ’dimensions’ can also be applied to the plot by determining the colour and point size, again determined by the data as shown in figure 6. The visualisation is special, in that any valid property of the options can be used as a graph dimension, and selected by a dropdown. Upon which the system

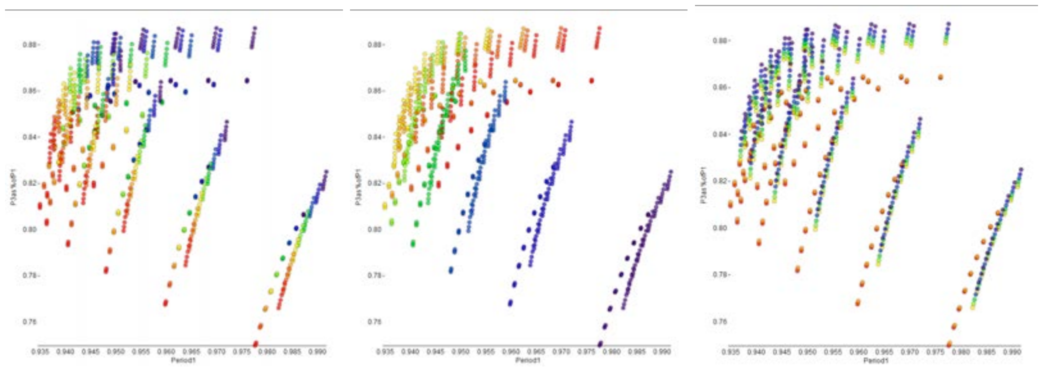


Figure 7: Visualisation showing coloration based on beam section family, shows that the 2nd family has the biggest influence on Period 1 (the X axis), and the 3rd family has the biggest impact on how similar period 3 is to period 1.

transitions the points to new configuration. So the various comparisons could be generated; P1 against P2, P2 against P3 etc. In this way trade-offs can be seen, and Pareto fronts of non-dominated options visualised.

In this case however there was interest in more than just a 2D trade-off. To enable this one solution used linked another plot, allowing for the Pareto front of one plot to be selected, showing up as highlighted points in the other. With any selected point being shown summary data on hover with the mouse, or if clicked then in a separate sidebar or window for deeper consideration.

In this case, the modal behaviour was not the only factor to consider, material usage was also important. To represent this the area of the circular point can be used to show the material volume. Presenting an informed trade-off where the material required for the given behaviour could be quickly and visually included in the decision making process.

An additional level of understanding can be realised by using colour to represent the section used for a given family. Although limited to one family, if viewed side by side as shown in figure 7, the change in colour with respect to the change in position in either axis shows a correlation (or not) to that factor.

By showing data in this way for easy comparison of many options, it is felt that not only can a better performance decision be made, but also the system can be better understood helping improve the intuition of the engineer.

5.2 Parametric Roof Design

This next project example represents a much more open design problem. It is concerned with the design of a canopy roof, which was defined parametrically. The roof had five major variations; roof height, grid size in X and Y, diagonal bracing spacing and element cross-section. All of which affect the engineering performance as well as have an impact on the architecture of the design. Some engineering requirements are necessary to ensure design safety however other represents trade-offs between engineering efficiency and visual preference. This is a typical state of affairs in structural design of buildings. As a result it is useful to look into new ways of understanding this and making the design decisions.

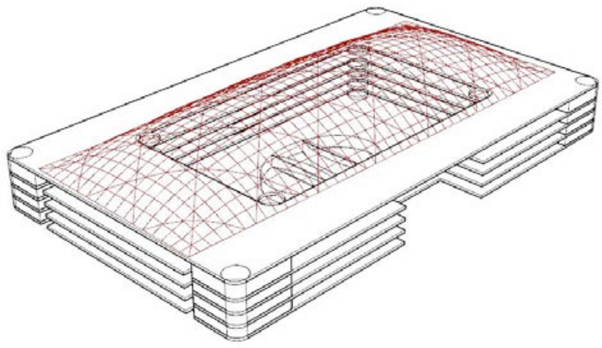


Figure 8: Basic design configuration, showing one variation of height, grid size, diagonal spacing and element cross-section.

In this case a large range of options was generated and analysed in a similar fashion as the previous example. In this case the number of design parameters is higher being five and so the samples of each was reduced to 3 beam sections, 4 bay sizes for both X and Y separately, 3 diagonal spacing options and 3 roof heights, resulting in 432 options. Each option was automatically generated analysed using a mixture of parametric modelling system Grasshopper coupled with batch processing in Python, with key data saved to a basic text file.

Initially the same graphing methods were used on the previous example. Made possible by the web page being developed to visualise as coordinates any numerical fields of a text file formatted with comma separated values. This enables a quick overview of data. For engineering concerns this highlights good trade-offs between significant performance requirements in this case deflection and material usage were key factors. However without any more data this limits decision support, to those based solely on engineering metrics, and obfuscates any understanding of the system.

So by referring to Bertin's visual properties [2], the visualisations can be modified to include more information, not just colour and size, but also circle stroke width and colour, even point transparency,

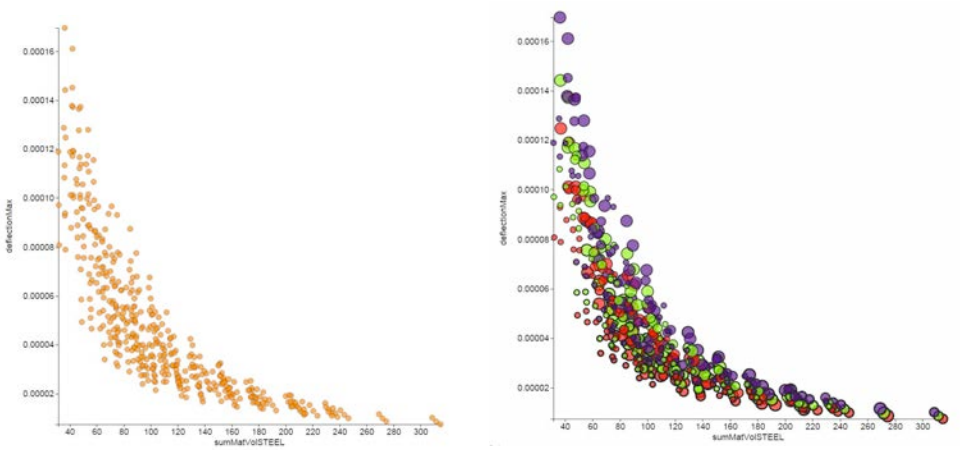


Figure 9: Left, Simple plot of deflection against material volume for a range of designs, as compared to the same graph right which also represents design inputs; colour : roof height, circle-radius : diagonal spacing, circle-stroke : section-thickness. All of these data visualisation associations can be changed by the user via the web interface.

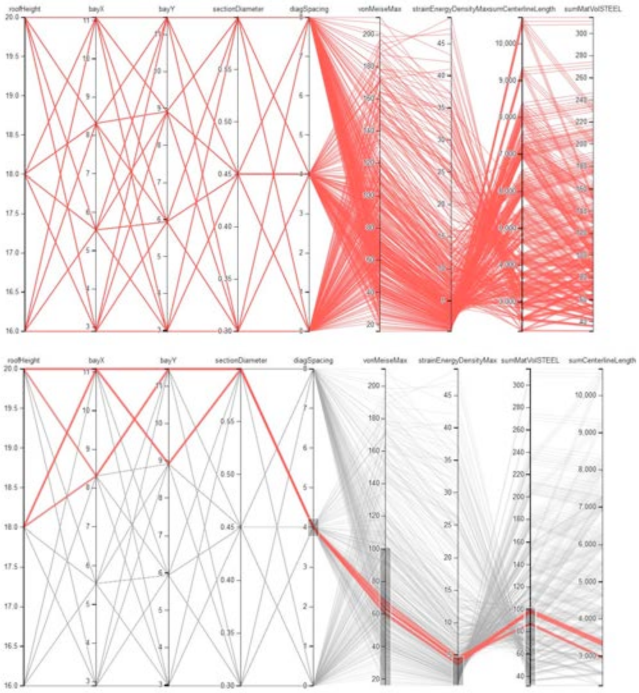


Figure 10: Parallel plot visualisation of roof design problem, with design parameters on the left and performance metrics on the right. Below a demonstrates a dynamic selection of design decisions and performance requirements by interactive sliders and the resultant compliant design options highlighted.

providing more visual degrees of freedom to display this multi-dimensional data as shown in figure 9.

This can be useful for inferring relationships between functional requirements (shown positionally) and design variables (shown by the points graphically). For example it can be seen that red (higher) roofs where closer to the Pareto front and so are preferable options. There is a danger that this can also become too busy for efficient understanding. Furthermore arguably it does not lend itself to choosing exploration of choosing options, which is often based on the interplay between engineering *needs* and architectural *wants*.

To address this issue another visualisation was built, again using D3. This uses a parallel plot, taking each dimension of the input design parameters and output analysis results and displaying it on a series of liner axis connecting each designs values with a line between the axis as shown in figure 10.

From here it is possible to allow the selection of specific designs based on ranges of one or more of the axis. By ordering input parameters on the left and analysis results on the right it is possible to generate interesting sets based on both functional and visual desires. For example by setting a maxim stress criteria, a minimum material range and a specific desirable cross bracing spacing a limited number of designs are available with usefully limit the ranges of other input parameters. This is of use in a design context for the implications of different design decisions to be made can be directly shown to have an impact on both engineering performance of a design but also limiting design variably in other regions.

6. Discussion

The previous examples have shown what can be developed in limited time based on real design data. However they have shown a significant impact on the depth and clarity of understanding of these problems. These methods can be made as specific or generic as the user requires, both of the previous examples take generic numerical data and are essentially agnostic of the problem. However it is felt by the author that the domain of structural engineering is well defined enough, that it would benefit from web visualisations specifically designed for the typical tasks undertaken, as these could be used on multiple projects successfully. Not just for option decision making but also more rudimentary single design appraisal, where a dashboard of relevant data could more efficiently convey the suitability of a model than the current tabular form.

Furthermore if developed specifically for engineering decision making problems, then a series of relevant visualisation pages or data-views could be linked together. With greater interactivity built in data so as data could be explored at different depths quickly in a similar way so one browses topics online on Wikipedia. For example selected options in the graphs above could open up new pages with more detail on the individual model, rather than just side bar information as shown here. This could be extended to include visualisation of the model also, using modern 3D web visualisation such as three.js, which could enable a complete decision making ecosystem.

It is important to note that the projects visualisations worked using just simple free web browser software. This allows it to work across platforms even on mobile devices, without requiring a potentially expensive FEM software licence. Resulting in a more efficient use model of software and hardware, with one set used to undertake the specialist task of analysis and acting a servers of data, which can then be distributed and consumed more easily by another set for visualisation and decision making.

7. Conclusions

This paper has highlighted some of the failings in data visualisation, both with modern monolithic engineering analysis software, but also the current typical spreadsheet based post processing. It proposes that a more appropriate paradigm would be to decouple analysis and visualisation, enabling better tools based on web technologies to be used in their place.

Some examples have been shown which leverage the interactive nature of web data visualisations to enable more information to be presented opening up potentials to compare numerous design options with greater ease. This is believed to be significant as it addresses the issues that parametric design presents by making option exploration easier and thus challenging existing engineering decision-making methodologies. Whilst these are simple examples and their power is hard to convey in an academic paper, they represent how easily such visualisations can be produced in this way as single webpages. Furthermore these are easily extended using all the inbuilt graphics and user interface capabilities of modern we browsers.

The field of data visualisation is a growing one with and with new frameworks the barriers to entry now significantly lower, with skills being equivalent to those used for existing spread sheet programming. As such it is believed by the author that this is something that design engineers should put more effort into adopting.

Acknowledgement

The author would like to acknowledge the support of Foster + Partners Applied Research and Development Group as well as EPSRC funded Industrial Doctorate Centre in Systems at The University of Bath (Grant EP/G037353/1).

References

[1] Aish, R., Fisher, A., Joyce, S., & Marsh, A. (2012). Progress towards Multi-Criteria Design Optimisation using DesignScript with SMART Form, Robot Structural analysis and Ecotect building performance analysis. ACADIA.

[2] Bertin, J. (1983). Semiology of graphics: diagrams, networks, maps.

[3] Bostock, M., Ogievetsky, V., & Heer, J. (2011). D³ data-driven documents. Visualization and Computer Graphics, IEEE Transactions on, 17(12), 2301-2309.

[4] Snow, John. On the mode of communication of cholera. John Churchill, 1855.

[5] McCandless, D. (2012). *Information is beautiful*. Collins.

[6] Minard, C. J. (1862). Des Tableaux graphiques et des cartes figuratives, par M. Minard,... Thunot.

[7] Playfair, W. (1786). Commercial and political atlas: Representing, by copper-plate charts, the progress of the commerce, revenues, expenditure, and debts of England, during the whole of the eighteenth century. London: Corry.

[8] Priestley, J. (1765). A chart of biography, London. British Library, London.

[9] Tufte, E. R., & Graves-Morris, P. R. (1983). The visual display of quantitative information (Vol. 2). Cheshire, CT: Graphics press.

[10] Tufte, E. R. (1997). Visual and statistical thinking: Displays of evidence for decision making. Graphics.

Integrated Design in the Simulation Process

Martha Tsigkari1, Angelos Chronis1, Sam Conrad Joyce1, Adam Davis1, Shuai Feng1, Francis Aish1

[1]Foster + Partners
Riverside, 22 Hester Road,
London, United Kingdom, SW11 4AN
mtsigikar, achronis, sjoyce, addavis, sfeng, faish @fosterandpartners.com

Keywords: Integrated Design, Integrated Practice, Simulation, Fast Feedback, Optimization, Environmental Analysis, FFD, Structural Analysis, View Analysis, Speed-Accuracy Trade-Off, Custom Interfaces, Multidisciplinary Team, Tool Development.

Abstract

During the past decade the construction industry has been witnessing a constant shift in the way it operates. The advances of technology have made possible the adaptation of a more direct, performance-driven design approach based on multi-objective - and sometimes contradicting – criteria of environmental, structural, economic and aesthetic impact. As a consequence, the various teams of consultants involved in the process no longer inform it consecutively, forcing various changes at different stages of the design. Instead, building projects increasingly comprise numerous design issues that can be delegated to small groupings of architects, engineers, and consultants to be resolved simultaneously, in parallel. In the light of this new *status quo,* the significance of new customized simulation tools and interfaces, capable of providing near real-time feedback and driven by multiple input criteria, looms as a potential game changer to the industry. This paper outlines the advances implemented by the authors to support these new integrated workflows.

1. INTRODUCTION

Building design is informed by diverse professions and criteria, most of which are vital in the way buildings perform, are perceived and experienced. Continuous advances in technology have allowed for digital simulation and verification of processes that until recently were based on manual calculations and rules of thumb. Consequently, contemporary software can accurately calculate the performative impact of various criteria, which enables quantitative and qualitative comparisons.

Although the *idea* of a performance driven design process is straightforward, its actual implementation is rather less so. There are various problems in achieving a seamless design process that can operate under the aforementioned premises. How do all the disciplines come together in an integrated fashion? How can inputs so varying, and in cases contradicting, be presented, understood and drive the process of design? How can this happen in a timely fashion and, furthermore, in the context of extremely fast design cycles, when most of the aforementioned simulations are so time-consuming, that by the time they finish they have become obsolete due to the projects' progression fast pace? How can the dialogue be facilitated by legibly expressing the trade-off between multiple criteria in increasingly complex design briefs?

This paper attempts to demonstrate how answers to the above questions were sought in the context of a project within an integrated practice. It will analyze the type of multi-disciplinary problems in hand, and how performance driven design based on multi-objective criteria can be achieved with the aid of fast-feedback tools and interfaces. It will argue for the need of customization and programmatic capability. This is in order to achieve near real-time analysis and simulation, which can accommodate rapid design cycles, by taking into advantage a speed-accuracy trade off. Finally it will discuss the ways by which Research & Development becomes a facilitator in the process of obtaining, analyzing, quantifying and explaining the impact of different parameters in design, and thus enhancing integration within a multidisciplinary industry.

2. INTEGRATION, SIMULATION & INTERACTIVE INTERFACES

The challenges designers often face are usually characterized by the following aspects:

- the diversity of the problems in hand, where the multiplicity of the input parameters makes divergence to an optimal solution non-intuitive
- the rapid turnaround of the design cycles that requires fast-feedback analyses, usually not available from off-the-shelf software packages
- the intricacy of interpreting overlaying analytical results, their interaction and how they ultimately affect the design

The above aspects can be overcome within the context of an *integrated practice*, where new tools can be developed in order to provide near-real time analysis and simulation and innovative custom interfaces can assist to the interpretation of the results.

2.1. Context

This performance driven design process is presented in the development of a residential project in Bangalore, India, undertaken by an international integrated practice (Figure 1). The design brief along with the complex constraints of the site posed a number of different challenges to which the design team had to respond. Specific goals included:

- Increase the potential for natural ventilation
- Maximise daylight to facades and covered public space
- Maximise views of greenery
- Minimise overlooking
- Minimise material usage in structural performance

This diverse set of goals required a multi-disciplinary response to inform design decisions at early conceptual stages. Moreover space planning constraints and aesthetical considerations were also important drivers that informed the design response.

Figure 1. Artist's Impression of Project

2.2. Design Complexity and the Need for a Multilayered Response

The need for better building performance, as driven by increasingly demanding regulations, is inevitably raising the complexity of the design solutions that are sought by contemporary practices. The cross-disciplinarity of the objectives of an overall optimally performing solution calls for new approaches that exceed the current professional capabilities. The fragmentation of professional responsibilities and the segmentation of the design process are considered to pose significant limitations in achieving these new levels of building performance (Cantin et al 2012). In many cases the analytic approach and the juxtaposition of different technical expertise creates a complex and often conflicting field. The benefits of the integration of building performance simulation tools at early design stages has very often been argued for but still remains a challenge (Attia and De Herde 2011). Despite the continuous effort to tackle this challenge, through the development and incorporation of simulation methods in conceptual stage tools, the complexity of current architectural projects often renders impossible an analytic approach (Hanna et al 2010).

This evolving design complexity, with which the current architect is confronted, requires a higher level of overview of the various aspects of each design problem and as well a better integration of the relevant disciplines in the design process. Arguably the effort to incorporate more '*architectural friendly*' simulation tools, through the development of more efficient interfaces and interoperable models greatly facilitates this goal (Attia et al 2009). However in order to allow the architectural ingenuity to thrive and incorporate the intuition back in the equation, a current practice requires a more integrated approach.

2.3. Integrated Practice

The definition of an integrated practice is still very much developing, as more practices have adopted this approach as a means to improve the delivery of their projects. However what is common in most examples of such practices is an attempt to reduce the barriers and hurdles for communication between disciplines. This can be literal in terms of having team members of different professions in the same building working on the same design. It can also mean having more inter-disciplinary meetings with communal goal setting and exploration rather than parallel studies and results sharing. This is especially relevant at concept stage where many fundamental but often less well founded decisions are made. Integral, of course, to this model is the commercial alignment of the various disciplines by them being in the same company, with the intended consequence that they are working towards the same goals.

2.4. Simulation as a Decision Maker – Speed Accuracy Trade Off

Simulation is a key tool used by engineers and consultants in order to analyze and interpret the effect that various conditions have not only on a building, but also at urban scale. Although simulation is very powerful, there are some drawbacks that make it hard to use as a driver on fast design cycles. First of all, many off-the-shelf simulation tools are limited by the amount of time they require in order to run a simulation. Secondly, the results are often difficult to interpret by non-specialist in the particular domain analyzed. The latter becomes even more prominent when various simulations processed for different criteria are seen in conjunction to each other.

But based on the assumption that modeling and simulation of building physics phenomena entails a certain amount of complexity and uncertainty as it depends on noisy and often conflicting input data (Hong 2000) we could see the potential of a trade-off between speed and accuracy when running simulations to accommodate the need for quick results, especially at the initial stages of the design process. On this premise, results accurate enough can be acquired much faster and drive the scheme by helping the designers make more informed decisions, especially during initials stages of a project. To accommodate the above considerations, an array of innovative, fast feedback analysis and simulation tools was developed.

3. PERFORMANCE DRIVERS

The authors of this paper are part of an interdisciplinary research and development team within the architectural practice. Their professional backgrounds in architecture, engineering, art and computer science allowed them to both understand and evaluate the inputs provided or required by the different parties within an integrated practice spectrum. On that aspect, they developed tools that helped the engineers and consultants run fast analyses and simulations and present them to all interested parties in a way that could become drivers in the scheme design process.

3.1. Fast Fluid Dynamics

The assessment of the wind conditions in an urban scale using computational fluid dynamikcs (CFD) simulations is traditionally time consuming, thus their application is mostly restricted to the final stages of the design process. CFD simulations have yet to fully meet the expectations of designers despite the development of novel methods and the constant increase of computational resources (Chen 2009). This complexity of the physics involved largely justifies this resource cost but in many cases it is argued that a compromise in accuracy might be required to inform design decisions at early stages. In CFD literature it is generally suggested that the accuracy of the CFD simulation needs to be leveled against the turnaround time requirements of its application and the level of detail needed by the task at hand (Lomax et al 2001).

To overcome the time requirements of standard CFD packages in order for them to meet the design cycle timeframes, the authors have incorporated a less accurate but much faster CFD solver. The FFD (Fast Fluid Dynamics) solver is based on a less accurate numerical solver, introduced by Stam (1999) to the computer graphics industry. Despite its numerical dissipation, the solver’s accuracy has been assessed as adequate for certain cases (Zuo and Chen 2010) and has been already used in a number of studies (Chronis et al 2011, Chronis et al 2012). Its turnaround time far exceeds typical CFD applications, therefore allowing for quick feedback of design iterations.

The FFD solver was used to simulate the airflow around the massing for the two prevailing wind directions in 3D. The simulation and interface tools were custom developed to allow for quick interoperability with the design platform used. In conjunction with the speed of the FFD solver, this could enable a CFD study for every new cycle of the design process. The simple interface that was developed allowed the design team to directly interact with the results, thus giving them a better understanding of the effect of the different design options on the wind flow.

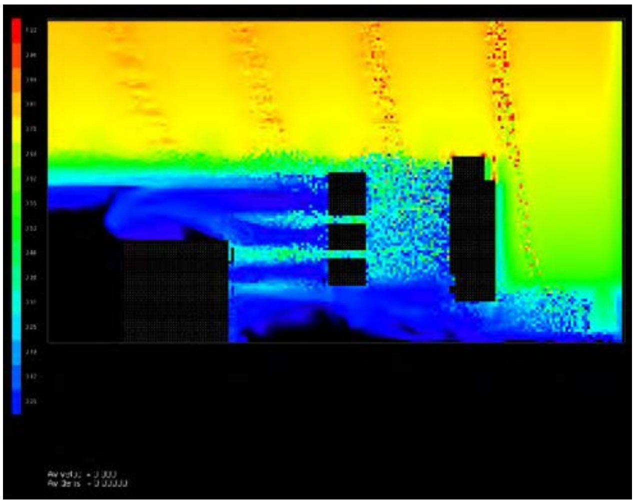


Figure 2. CFD Analysis viewed via a customized interface

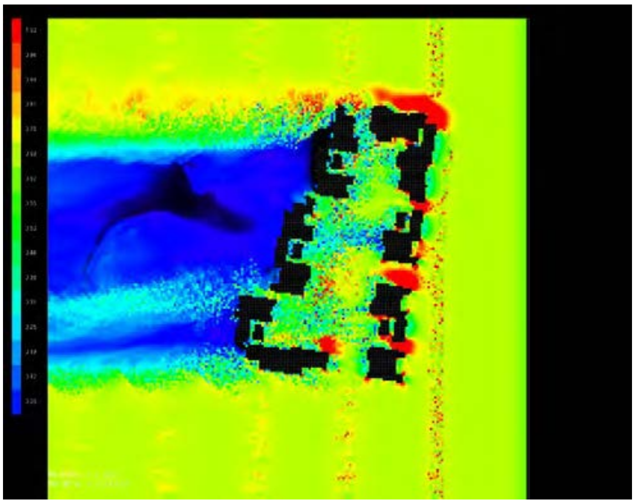


Figure 3. CFD Analysis viewed via a customized interface

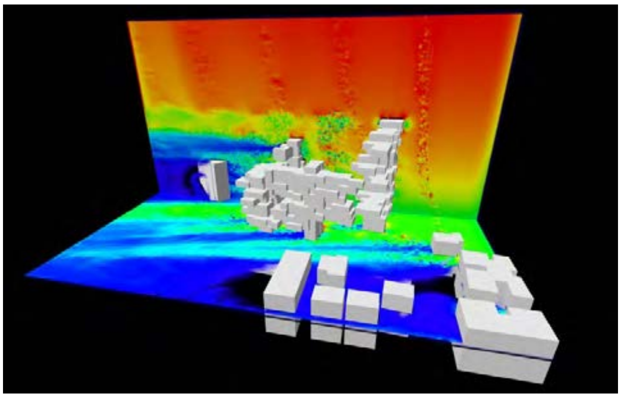


Figure 4. CFD Analysis viewed via a customized interface

3.2. Solar and Daylight Analysis

It has been argued that more precise solar design tools actually help to broaden the range of architectural form (Otis 2011). But research has showed that while there is a growing number of practitioners who dedicate a significant amount of time to daylighting, most have but a couple of days per project to spend on this topic, as reasons for not using simulations were mostly linked to time and budget constraint (Galasiu and Reinhart 2008). Furthermore, the time needed to run many existing environmental software tools is increased due to poor interoperability, which requires timely file conversions to adequate formats for analysis. Based on this aspect a need was identified for a new and powerful tool, that can provide accurate and fast solar and daylight analysis, within the software platform used in-house.

This new near-real-time solar and daylight analysis software (called RadIO for Radiance I-O), as Diva before it (Lagios et al. 2010), was developed using the Radiance ray trace engine (Ward and Shakespeare 1998; Mardaljevic 1995), as well as the GenCumulativeSky method (Robinson and Stone 2004) and is capable of calculating seasonal radiation maps, daylight factor and vertical sky component (Chronis et al 2012). Furthermore, a set of extra capabilities were exposed in a user-friendly manner, such as the calculation of sunlight hours, relative to different local codes. This tool offers tremendous performance gains, as it is orders of magnitude faster than other commercial software (Chronis et al 2012).

Using the RadIO tool, many massing options could be run natively in the office design platform (Microstation), and very quickly provide comparative feedback. It proved very powerful in comparing completely different options and ranking them against each other. It demonstrated that the more complex massings were performing better than simpler ones, a fact that was somewhat counter-intuitive and hard to prove otherwise (Figures 5, 6).

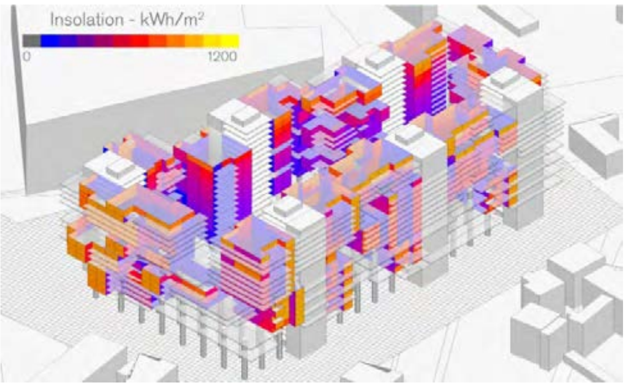


Figure 5. Insolation Analysis run inside Microstation via RadIO

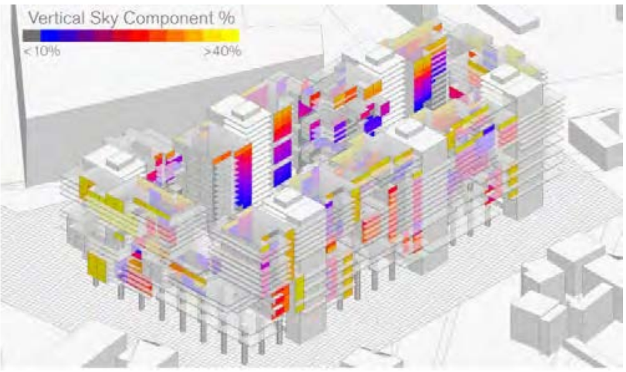


Figure 6. VSC Analysis run inside Microstation via RadIO

3.3. Reflection and Soffit Analysis

In certain situations analytical processes come to bridge the disconnection between conceptual ideas and their realization within the specific site and economic constrains. Within an integrated design process, not only engineering and optimization aspects can be integrated into decision making, but moreover key improvements on the design concept can be implemented through comprehensive parametric strategies.

Under that spectrum, and progressing from the general daylight potential analysis, the next challenge was the development of a soffit that would emerge into a tree-like pattern, recalling the memory of the local grown forest, while providing extra performances such as shading, reflective lighting and even evaporative cooling. The design task was initially incarnated into a conceptual drawing, from which the soffit of the building would generatively emerge in a standarized and performative manner. The general idea incorporated the use of a finite number of standardized reflective tiles that can create a tree patterns on the soffit, with the columns as their notional trunks.

For the development of the parametric analytical tool, the emphasis was on reflectivity of the pattern as well as buildability and standardization of the components, promoting economy in material and construction costs. That was a response to the architects' expectation of a trade-off between a aesthetically driven pattern generation process, and affordability for local available traditional handcraft techniques.

The analytical approach of the tool can be understood both in the pattern input selection (which is then tessellated based on standard components) as well as its diffuse light capability performance. On one hand, it leaves enough flexibility in terms of pattern inputs. Research undertaken together with the architects, delivered a selection pool of appropriate local tree canopy patterns which represent the appeal delivered through the architectural conceptual drawings. On the other hand, a patterns' arrangement study has been progressed alongside, to provide valid light reflection paths in relation to the locations of the 'reflectors' that have been housed underneath the soffit, such as highly reflective tile ground and water ponds.

Figure 7. Parametrically Generated Soffit Patterns

Eventually, a process of pixelation has been conducted to rationalize the complexity of the pattern selected (Figure 7). During this, the study of the light reflection paths has been taken into consideration, providing differentiated distributed densities of the mirror components, in order to facilitate a better reflected lighting environment underneath the soffit. Throughout the rationalization process, the feedback from the architects was taken into account. This is demonstrated in the flexibility left for the properties of the mirror components, such as their sizes, shapes, thicknesses, their numbers of types in gradient transitions and their positive or negative physical attributes. Visualization of one of the options can be viewed in Figure 8.

Figure 8. Suggested customised standard tiling of reflective soffit

3.4. Structural Analysis

When looking to engineer a design that is influenced by the many interacting factors as outlined, it was thought that a flexible method of generating the structural arrangement that provided meaningful feedback was required. It was important not simply to produce structural analysis, but actually go beyond this to give an indication of effective sizes of columns and the placements of shear walls to allow the team to access impactful feedback of the structural scheme. The goal being, that based on the initial configuration, an approximate but working structural scheme could be returned for appraisal alongside the other design considerations.

The task was divided into two optimization problems working at different resolutions. The first was the configuration of the cross-bracing, and the second the sizing of the members. The building being a basic framed grid with supporting shear wall bracing able to be placed anywhere in that grid, presented a high number of possible bays to brace and thus an even higher number of possible combinations of walls. For this reason a genetic algorithm approach was applied similar to that outlined in Evins, Joyce et al (2012). The coding genome of the design was a binary on / off of the cross bracing in the planar slice of the structure considered - essentially a 2D grid. The objective function to be optimized was linked to the output of the second optimization problem. For the second optimization a method was applied similar to that outlined in Joyce, Fisher et al (2011) where members where sized relative to their stresses in an iterative fashion. This provided a figure for the weight of the material required for the cross-bracing configuration considered by the first optimization to not structurally fail. This tonnage could then be passed back to the objective function of the first optimization and combined with terms to penalize over use of cross-bracing, which it was decided was disadvantageous to the overall design. The first optimization then after a number of generations produced optimized configurations of cross-bracing, still retaining the thickened structural elements from the second optimization. The output was presented for fast appraisal as a simple compression and tension diagram with the elements having exaggerated thickness to show the variation in thickness (Figure 9). This enabled designers to interpret the result quickly especially the means of spanning and placement of thickened members which were the primary designed part of the structure. Using this approach comparisons both quantitative (material volume) and qualitative (aesthetic) could be made between different amounts of cross-bracing used.

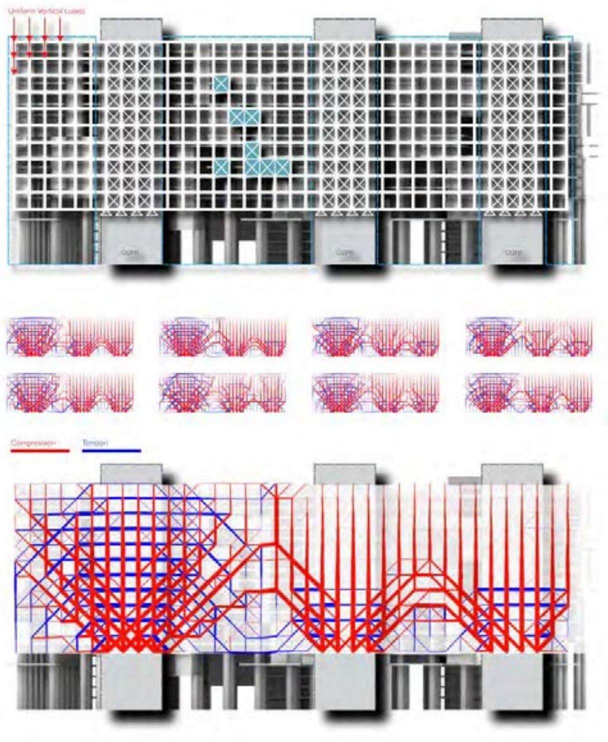


Figure 9. Structural Optimization through Evolution: An example of a optimized planar structure section for given boundary conditions

3.5. View Analysis

As the scale of an architectural project increases, so do the complexity of the issues which have a bearing on the visual apprehension and appreciation of the environment. Designers need to consider the effects of different schemes with regard to such criteria as views of varying depths, the effect of obstruction - and conversely, framing - of views of interest, and the visibility of 'natural' landscape elements (water, vegetation) if possible.

There is scant literature to establish any of these issues in such a way as to make them amenable to computational analysis based on objective criteria. Some of the authors of this paper are currently engaged in research seeking to establish perceptual bases for judgments about the visual environment. A first step in this is simply measuring visibility of subjects of interest (landscape, landmarks) and disinterest (nearby windows, where privacy may be a concern). Secondly, it is possible using image analysis techniques to understand something of the composition of a view: how segmented is a view of a landmark, or rolling hills in the distance? Given the orientation of a window, is a point of interest centered or oblique in the field of view?

In the context of the residential scheme that is the subject of this paper, it was important to assess the availability of views of green space and vegetation -- surrounding landscape, courtyard garden, and vegetated elements of facades within the scheme. Scenes were rendered from the facade of each apartment. Features of interest were extracted from these scenes and measured for the amount of area they occupied proportional to the total field of view. This metric of prevalence in the visual field is proposed to be a suitable reflection of view quality at least insofar as it reflects potential availability of features of interest in the scene. In the same way that vertical sky component measures potential daylight for a given facade area, this metric of view quality represents availability of interesting scene features on a facade. (Figure 10). Views from within architectural space have considerable qualitative importance, yet objective quantitative metrics can also be established that allow for design iterations to be analysed and optimised numerically, much like the other types of analysis discussed in this paper.

Figure 10. Quality of View Analysis for the Bangalore scheme

4. CONCLUSION

The implication that we find might pertain to our work in building design and analysis can be summed up -- perhaps ironically -- by decades-old mantras of UNIX systems:

- Write programs that do one thing and do it well
- Write programs to work together

Our experience suggests that 'doing it well' means producing results rapidly, to the level of precision appropriate to the stage of design. Large-scale, monolithic applications still have their place when evaluating a relatively static proposal over the course of days or weeks, but early-stage design requires agility on the part of decision-makers and the analysis tools available to them.

We find that composing modular tools that work in this fashion improves the ability of our software and our designers and analysts to work together. This allows not only for standard user-initiated analysis workflows, but also for computer-controlled routines such as optimization loops between different types of analysis. It also allows for assessment to be conducted regularly, as it is easy to evaluate the simulation data at any stage in the design and analysis process, rather than waiting for many batches or iterations to run before any judgement can be made.

Acknowledgements

We would like to thank our colleagues in the design and engineering teams with whom we collaborated on the case study project.

References

ATTIA, S. AND DE HERDE, A. 2011. Early design simulation tools for net zero energy buildings: a comparison of ten tools, International Building

Performance Simulation Association, November 2011, Sydney, Australia.

ATTIA, S., BELTRÁN L., DE HERDE, A, HENSEN, J. AND DE HERDE A. 2009. “Architect Friendly”: a comparison of ten different building performance simulation tools, International Building Performance Simulation Association, July 2009, Glasgow, UK.

CANTIN, R., KINDINIS, A. AND MICHEL, P. 2012. New approaches for overcoming the complexity of future buildings impacted by new energy constraints, Futures, Volume 44, Issue 8, October 2012, Pages 735-745.

CHITTKA, L., SKORUPSKI, P. AND RAINE N.E. 2009. Speed–accuracy tradeoffs in animal decision making. In Trends in Ecology & Evolution, vol. 24 (7): 400-407.

CHRONIS, A., TURNER, A. AND TSIGKARI, M. 2011. Generative Fluid Dynamics: Integration of Fast Fluid Dynamics and Genetic Algorithms for wind loading optimization of a free form surface. Proceedings of the Symposium on Simulation for Architecture and Urban Design at the 2011 Spring Simulation Multiconference, Boston, USA.

CHRONIS, A., TSIGKARI, M., DAVIS A. AND AISH, F. 2012. Design Systems, Ecology + Time. Proceedings of ACADIA at the 2012 ACADIA Conference, San Francisco, USA.

EVINS R, JOYCE S, POINTER P, SHARMA S, VAIDYANATHAN R, WILLIAMS C, 2012. Multi-objective design Optimization: getting more for less Proceedings of the ICE - Civil Engineering, Volume 165, Issue 5, 01 May 2012 , pages5 –10

GALASIU A. AND C. REINHART. 2008. Current Daylighting Design Practice: A Survey. In Building Research and Information, Vol. 36 (2), 159-174. London: Routledge

HANNA, S., HESSELGREN, L., GONZALEZ, V. AND VARGAS, I. 2010. Beyond Simulation: Designing for Uncertainty and Robust Solutions. Proceedings of the Symposium on Simulation for Architecture and Urban Design at the 2010 Spring Simulation Multiconference, Orlando, USA.

HONG, T., CHOU, S.K. AND BONG T.Y. 2000. Building simulation: an overview of developments and information sources. In Building and Environment, vol 35 (4): 347-361.

JOYCE S, FISHER A, WILLIAMS C, SHARMA S. 2011. Towards Ubiquitous Structural Frame Design Tools, IASS IABSE Conference 2011, London. UK. .

LOMAX, H., PULLIAM, T. H., ZINGG, D. W., PULLIAM, T. H. AND D. W. ZING. 2001. Fundamentals of computational fluid dynamics. Berlin: Springer..

LAGIOS, K., NIEMASZ, J. AND REINHART, C.F. 2010. Animated Building Performance Simulation (ABPS) - Linking Rhinoceros/Grasshopper with Radiance/Daysim. Proceedings of SimBuild 2010. New York.

OTIS T. 2011. Solar design a straightjacket for architecture? Proceedings of PLEA Conference 2011. Boston: Passive and Low Energy Architecture.

MALKAWI, A.M. 2004. Developments in environmental performance simulation. Automation in Construction 13(4). Pages 437-445.

MARDALJEVIC, J. 1995. Validation of a Lighting Simulation Program under Real Sky Conditions. In Lighting Research & Technology, vol 27(4): 181-188.

ROBINSON, D. AND A. STONE. 2004. Irradiation modeling made simple: The cumulative sky approach and its applications. Proceedings of PLEA Conference 2004: 1-5. Eindhoven: Passive and Low Energy Architecture.

STAM 1999. Stable fluids in Proceedings of the 26th annual conference on Computer graphics and interactive techniques, August 1999, Los Angeles, USA, 121-128.

ZUO, W. AND CHEN, Q. Y. 2007. Validation of fast fluid dynamics for room airflow, Proceedings of the 10th international IBPSA conference Building Simulation 2007, September 2007, Beijing, China.

WARD, G. AND R. SHAKESPEARE. 1998. Rendering with RADIANCE, The Art and Science of Lighting Visualization. San Francisco: Morgan Kaufmann Publishers.

The use of a particle method for the modelling of isotropic membrane stress for the form finding of shell and fabric structures

Francis Aish[a], Sam Joyce[a], Samar Malek[b], Chris J K Williams[b,1]

[a]Foster + Partners, Riverside, 22 Hester Road, London SW11 4AN
[b]Department of Architecture and Civil Engineering, University of Bath, Bath BA2 7AY

Abstract

The best known isotropic membrane stress state is a soap film. However, if we allow the value of the isotropic stress to vary from point to point then the surface can carry gravity loads, either as a hanging form in tension, or as a masonry shell in compression. The paper describes the theory of isotropic membrane stress under gravity load and introduces a particle method for its numerical simulation for the form finding of shell and fabric structures.

Keywords: masonry shell, isotropic stress, minimal surface, particle methods

1. Introduction

Masonry shells can only work in compression and a number of techniques have been developed for finding their geometry to achieve a specified stress state [1, 2, 3, 4]. In this paper we propose the use of a variable isotropic stress state where the membrane stress is uniform in all directions with no shear stress, but the value of the stress varies from point to point.

There is no particular reason why the compressive membrane stress should be isotropic, but it could be argued that an isotropic stress is in some ways optimum. This mirrors the argument that a minimal surface is the best shape for a cable net or fabric structure.

We will use the expression surface tension to denote the value of the isotropic membrane stress expressed as a force per unit length. If the stress

[1]Corresponding author

is compressive, then the surface tension is negative. It is often thought that the surface tension in a soap film is constant, but if this were the case it would not be possible for a soap film to carry its own weight. A vertical soap film must have a higher surface tension at the top than at the bottom. The situation is analogous to hydrostatic pressure that must increase with depth.

Even without gravity surface tension must vary with film thickness. Imagine a soap film with slight fluctuations in thickness. The surface tension must be greater where the film is thinner to pull fluid from thicker areas to ensure stability [5].

In the following sections we present the theoretical analysis of an isotropic membrane stress under gravity loads. We then formulate and illustrate the use of particle methods for the numerical simulations.

2. Theoretical analysis

2.1. Geometric preliminaries

The methods described in sections 2.1 and 2.2 are based on those in Green and Zerna[6], but with some changes in notation.

Consider a surface described by the position vector

$$\mathbf{r}\left(\theta^1,\theta^2\right) = x\left(\theta^1,\theta^2\right)\mathbf{i} + y\left(\theta^1,\theta^2\right)\mathbf{j} + z\left(\theta^1,\theta^2\right)\mathbf{k}. \tag{1}$$

$\mathbf{i}$, $\mathbf{j}$ and $\mathbf{k}$ are unit vectors in the directions of the Cartesian axes and θ^1 and θ^2 are the surface parameters or coordinates replacing the u and v which are often used. Note that the 1 and 2 are not exponents.

The covariant base vectors are

$$\mathbf{g}_i = \frac{\partial \mathbf{r}}{\partial \theta^i} = \frac{\partial x}{\partial \theta^i}\mathbf{i} + \frac{\partial y}{\partial \theta^i}\mathbf{j} + \frac{\partial z}{\partial \theta^i}\mathbf{k} \tag{2}$$

in which i is equal to 1 or 2. $\mathbf{g}_1$ and $\mathbf{g}_2$ are tangent to the surface in the directions of increasing θ^1 and θ^2 respectively. Note that they are in general not unit vectors, nor are they perpendicular to each other.

The components of the metric tensor are

$$g_{ij} = \mathbf{g}_i \cdot \mathbf{g}_j \tag{3}$$

and the square of the distance between adjacent points on the surface is

$$\delta s^2 = \left(\sum_{i=1}^{2} \frac{\partial \mathbf{r}}{\partial \theta^i}\delta\theta^i\right) \cdot \left(\sum_{j=1}^{2} \frac{\partial \mathbf{r}}{\partial \theta^j}\delta\theta^j\right) = \sum_{j=1}^{2}\sum_{i=1}^{2} g_{ij}\delta\theta^i\delta\theta^j = g_{ij}\delta\theta^i\delta\theta^j. \tag{4}$$

The summations in the right hand side of this expression are implied by the Einstein summation convention. This expression for δs^2 is known as the first fundamental form and therefore g_{ij} are also known as the coefficients of the first fundamental form.

The quantity

$$g = g_{11}g_{22} - g_{12}^2 \tag{5}$$

and the unit normal,

$$\mathbf{n} = \frac{\mathbf{g}_1 \times \mathbf{g}_2}{|\mathbf{g}_1 \times \mathbf{g}_2|} = \frac{\mathbf{g}_1 \times \mathbf{g}_2}{\sqrt{g}}. \tag{6}$$

Note that g is not a scalar in that it is a property of the coordinate system, rather than something with physical meaning.

The contravariant base vectors $\mathbf{g}^j$ also lie in the plane of the surface. They are defined by

$$\begin{aligned} \mathbf{g}_i \cdot \mathbf{g}^j &= \delta_i^j \\ \mathbf{n} \cdot \mathbf{g}^j &= 0 \end{aligned} \tag{7}$$

in which the Kronecker deltas, $\delta_i^j = 0$ if $i \neq j$ and $\delta_i^j = 1$ if $i = j$. Thus $\mathbf{g}^1$ is perpendicular to both $\mathbf{g}_2$ and $\mathbf{n}$ and its magnitude is such that $\mathbf{g}_1 \cdot \mathbf{g}^1 = 1$.

The contravariant components of the metric tensor are

$$g^{ij} = \mathbf{g}^i \cdot \mathbf{g}^j \tag{8}$$

and a vector can be expressed as

$$\mathbf{v} = v^i \mathbf{g}_i + v\mathbf{n} = v_i \mathbf{g}^i + v\mathbf{n} \tag{9}$$

in which

$$\begin{aligned} v^i &= g^{ij} v_j \\ v_i &= g_{ij} v^j. \end{aligned} \tag{10}$$

Again note the use of the summation convention in (9) and (10).

Finally, the coefficients of the second fundamental form are

$$b_{ij} = b_{ji} = \frac{\partial \mathbf{g}_i}{\partial \theta^j} \cdot \mathbf{n} = \frac{\partial \mathbf{g}_j}{\partial \theta^i} \cdot \mathbf{n} = -\mathbf{g}_j \cdot \frac{\partial \mathbf{n}}{\partial \theta^i} \tag{11}$$

and the second fundamental form itself is

$$\delta \mathbf{r} \cdot \delta \mathbf{n} = -b_{ij} \delta\theta^i \delta\theta^j. \tag{12}$$

b_{ij} tell us about how the direction of the normal changes as we move about on the surface, in other words, about the curvature of the surface.

b_{ij} and g_{ij} are not independent, they are linked by the Gauss Codazzi Mainardi equations which ensure that the surface fits together.

2.2. The membrane equilibrium equations for shell and tension structures

We are now in a position to define the membrane stress tensor $\boldsymbol{\sigma}= \sigma^{ij}\mathbf{g}_i\mathbf{g}_j$ by

$$\delta\mathbf{f} = \epsilon_{ik}\sigma^{ij}\mathbf{g}_j\delta\theta^k \tag{13}$$

in which $\delta\mathbf{f}$ is the element of force crossing the imaginary cut $\delta\mathbf{r} = \mathbf{g}_k\delta\theta^k$. $\epsilon_{12} = -\epsilon_{21} = \sqrt{g}$ and $\epsilon_{11} = 0$ and $\epsilon_{22} = 0$ are the components of the Levi-Civita permutation pseudotensor. Note that we are not yet making the assumption that the membrane stress is isotropic.

Equation (13) makes a bit more sense when written out in full:

$$\delta\mathbf{f} = \sqrt{g}\left(\sigma^{11}\delta\theta^2 - \sigma^{21}\delta\theta^1\right)\mathbf{g}_1 + \sqrt{g}\left(\sigma^{12}\delta\theta^2 - \sigma^{22}\delta\theta^1\right)\mathbf{g}_2, \tag{14}$$

especially when compared to the equivalent relationship for plane stress in two dimensions in Cartesian coordinates:

$$\delta\mathbf{f} = (\sigma_x\delta y - \tau_{yx}\delta x)\,\mathbf{i} + (\tau_{xy}\delta y - \sigma_y\delta x)\,\mathbf{j}. \tag{15}$$

Equilibrium of moments about the surface normal tell us that the stress tensor is symmetric, $\sigma^{12} = \sigma^{12}$.

Adding the forces on a small quadrilateral of shell we have

$$\frac{\partial}{\partial\theta^2}\left(\epsilon_{i1}\sigma^{ij}\mathbf{g}_j\left(-\delta\theta^1\right)\right)\delta\theta^2 + \frac{\partial}{\partial\theta^1}\left(\epsilon_{i2}\sigma^{ij}\mathbf{g}_j\delta\theta^2\right)\delta\theta^1 + \mathbf{w}\sqrt{g}\delta\theta^1\delta\theta^2 = 0 \tag{16}$$

where $\mathbf{w}$ is the load per unit surface area. Thus

$$\frac{\partial}{\partial\theta^i}\left(\sqrt{g}\sigma^{ij}\mathbf{g}_j\right) + \mathbf{w}\sqrt{g} = 0. \tag{17}$$

In terms of components this can be written as

$$\sigma^{ij}b_{ij} + w = 0 \tag{18}$$

which is the equilibrium equation in the direction of the normal and

$$\nabla_i\sigma^{ij} + w^j = 0 \tag{19}$$

which are the two equilibrium equations in the plane of the surface. $\nabla_i\sigma^{ij}$ is the covariant derivative,

$$\nabla_i\sigma^{ij} = \frac{\partial\sigma^{ij}}{\partial\theta^i} + \sigma^{kj}\Gamma^k_{ki} + \sigma^{ik}\Gamma^j_{ki} \tag{20}$$

and

$$\Gamma^j_{ki} = \mathbf{g}^j\cdot\frac{\partial\mathbf{g}_k}{\partial\theta^i} \tag{21}$$

are the Christoffel symbols of the second kind.

2.3. The membrane equilibrium equations for isotropic membrane stress shells

The previous section provided the theoretical background for understanding the membrane equilibrium equations for shell and tension structures. Next, we derive the equations accounting for isotropic membrane stress.

If the state of stress is isotropic the membrane stress tensor is

$$\sigma^{ij} = \sigma g^{ij} \tag{22}$$

in which the scalar σ is the surface tension with units force per unit width.

The equilibrium equations now become

$$\sigma g^{ij} b_{ij} + w = 0 \tag{23}$$

and

$$\nabla_i \left(\sigma g^{ij}\right) + w^j = 0. \tag{24}$$

However

$$g^{ij} b_{ij} = 2H \tag{25}$$

where H is the mean or Germain curvature so that in the normal direction,

$$2\sigma H + w = 0. \tag{26}$$

When the loading is zero we have $H = 0$ which is the condition for a minimal surface.

The covariant derivatives of the components of the metric tensor are zero so that the in-plane equilibrium equations become

$$\frac{\partial \sigma}{\partial \theta^i} + w_i = 0. \tag{27}$$

2.4. Vertical loading on isotropic membrane stress shells

If the loading is vertical then

$$\begin{aligned} w_i &= -W \mathbf{g}_i \cdot \mathbf{k} \\ w &= -W \mathbf{n} \cdot \mathbf{k} \end{aligned} \tag{28}$$

in which W is the downwards load per unit surface area. However,

$$\mathbf{g}_i \cdot \mathbf{k} = \frac{\partial z}{\partial \theta^i}, \tag{29}$$

so that the in-plane equilibrium equations become

$$\frac{\partial \sigma}{\partial \theta^i} = W \frac{\partial z}{\partial \theta^i}. \tag{30}$$

Thus σ must be a constant along a contour line of constant z and therefore σ must be a function of z only. Thus

$$\frac{d\sigma}{dz} = W \tag{31}$$

and W is therefore also a function z only.

If we write

$$W = W(\sigma) \tag{32}$$

where $W(\sigma)$ is a function that we have chosen, then

$$z = \int \frac{d\sigma}{W} \tag{33}$$

giving us the relationship between W, σ and z.

The equilibrium in the normal direction is

$$2\sigma H = W \cos \lambda \tag{34}$$

where λ is the slope of the shell, that is the angle between $\mathbf{n}$ and $\mathbf{k}$.

2.5. Constant physical stress shell

The physical stress in a shell of thickness t is

$$\sigma_{\text{physical}} = \frac{\sigma}{t} \tag{35}$$

in units of force per unit area. Remember that σ is the membrane stress with units force per unit length. The weight per unit surface area is

$$W = \rho g t$$

where ρ is the density and g is the acceleration due to gravity (not to be confused with the geometric quantity with the same symbol).

Thus

$$\frac{\sigma_{\text{physical}}}{\rho g} = \frac{\sigma}{W} \tag{36}$$

and therefore if the ratio $\sigma_{\text{physical}}/\rho$ is constant,

$$\frac{\sigma}{W} = -a \tag{37}$$

in which a is a constant with units of length. The minus sign is to give us a negative σ corresponding to compression when W and a are positive.

Then

$$\frac{d\sigma}{dz} = -\frac{\sigma}{a} \tag{38}$$

and therefore the thickness,

$$t = t_0 \exp\left(\frac{z_0 - z}{a}\right). \tag{39}$$

The shape of the shell is given by

$$2aH + \cos\lambda = 0 \tag{40}$$

in which H is negative for a dome-like shell.

2.6. Weight per unit area a linear function of z

If

$$W = W_0 + Q\,(z_0 - z) \tag{41}$$

where Q is a constant, then

$$\sigma = \sigma_0 - \frac{W^2 - W_0^2}{2Q}, \tag{42}$$

unless $Q = 0$ in which case W is a constant and

$$\sigma = \sigma_0 - W\,(z_0 - z)\,. \tag{43}$$

We will focus our attention on the case when

$$\sigma_0 = -\frac{W_0^2}{2Q} \tag{44}$$

so that

$$\sigma = -\frac{W^2}{2Q} \tag{45}$$

and

$$\frac{\sigma}{W} = -\frac{W_0}{2Q} - \frac{1}{2}\left(z_0 - z\right). \tag{46}$$

The membrane stress is proportional to the thickness squared, rather than just the thickness as in the previous section. This is justified by the following reasoning. We know that the linear buckling theory of shells can give extremely optimistic results, but the Zoelly and Van der Neut formula [7] tells us that the linear buckling load of a spherical shell of given radius is proportional to the square of the thickness. This is because buckling involves bending stiffness as well as membrane stiffness [8]. Clearly non-linear buckling is also influenced by both bending and axial stiffness.

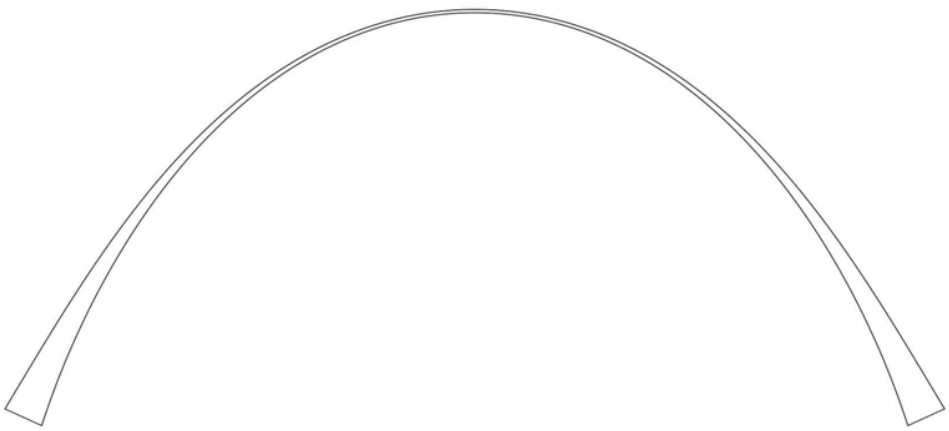


Figure 1: Shell with constant physical stress

2.7. Shells of revolution

In cylindrical polar coordinates a shell of revolution is described by $z = z(r)$ and the slope λ and arc length along the cross-section s are related to r and z by $dr/ds = \cos\lambda$ and $dz/ds = \sin\lambda$.

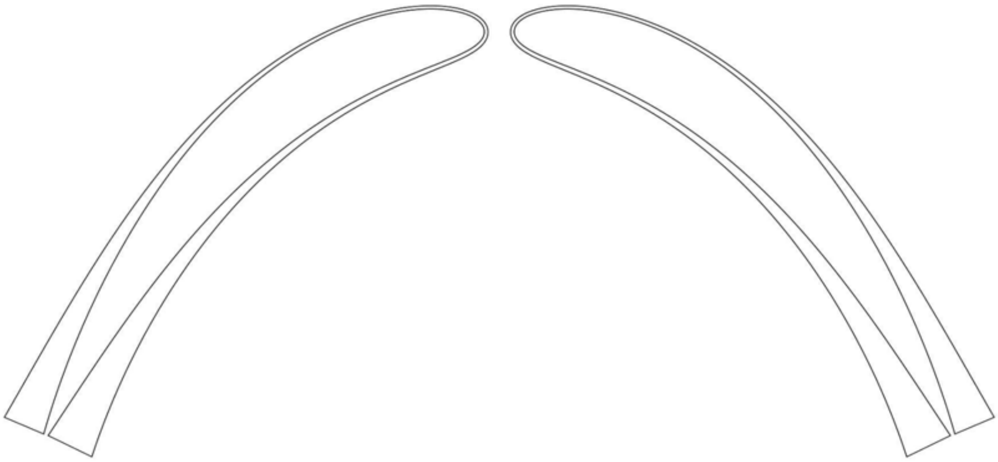


Figure 2: Shell with constant physical stress and oculus

The mean curvature is

$$H = \frac{1}{2}\left(\frac{d\lambda}{ds} + \frac{\sin\lambda}{r}\right) \tag{47}$$

and therefore

$$\frac{d\lambda}{ds} = 2H - \frac{\sin\lambda}{r} = \frac{W}{\sigma}\cos\lambda - \frac{\sin\lambda}{r} \tag{48}$$

in which W/σ is a known function of z. There is an analytic solution to this equation for the case when $W = 0$, the catenoid minimal surface, $r = \cosh(z/c)$.

There is also the cone

$$\frac{\sigma}{W} = \frac{z}{2} \tag{49}$$

which satisfies (31) and produces

$$0 = \frac{d\lambda}{ds} = \frac{2}{z}\cos\lambda - \frac{\sin\lambda}{r} \tag{50}$$

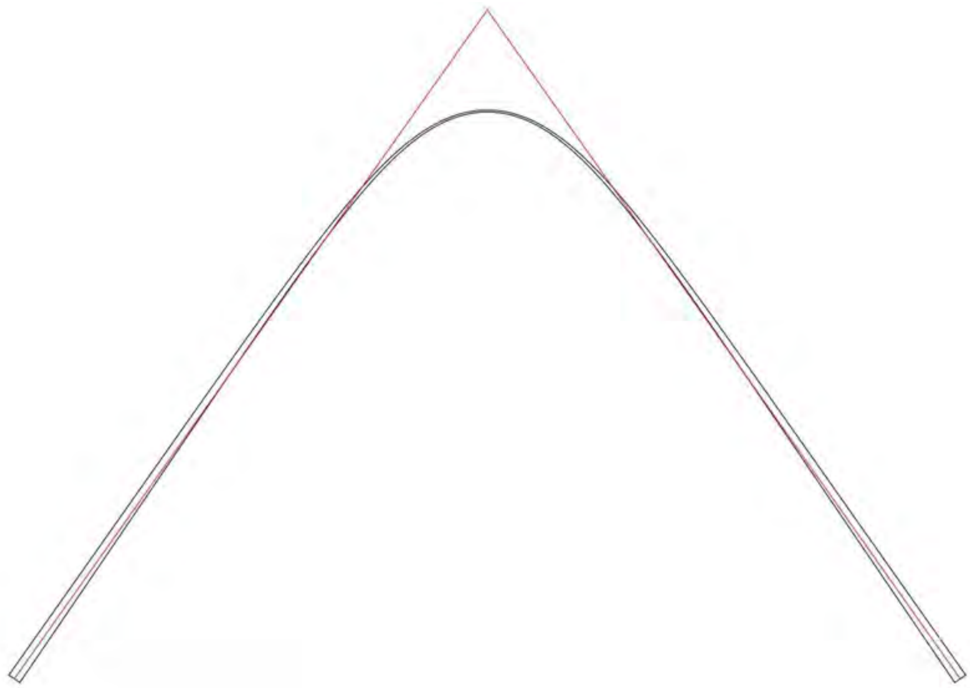


Figure 3: Shell with $\sigma/W^2 =$ constant

so that

$$\lambda = \tan^{-1}\left(\sqrt{2}\right) = 54.7°. \tag{51}$$

For other cases we can integrate numerically by marching along the curve. Figures 1 and 2 show the uniform physical stress shell, $\sigma/W =$ constant, while figures 3 and 4 show the $\sigma/W^2 =$ constant shell. In each case there is the possibility of a shell closed at the top or a shell with an oculus surrounded by a catenoid-like section. Figure 3 shows a 54.7° slope cone in red.

3. Numerical form finding using a system of particles

Techniques for the numerical form finding of tension structures by modelling a soap film are well established, usually using flat triangular finite elements. The area of a triangle is half the base times the height, hence the forces that a triangle exerts on its nodes are equal to the surface tension times half the length of the opposite side acting in the direction perpendicular to the side [9]. However, we shall investigate the use of a particle method

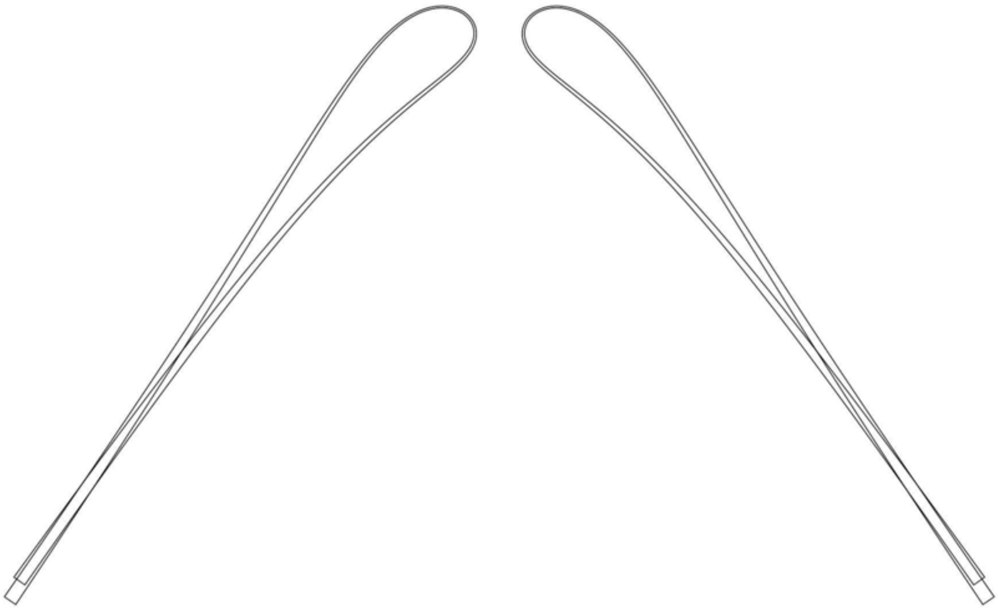


Figure 4: Shell with σ/W^2 = constant and oculus

using techniques developed for smoothed particle hydrodynamics [10]. This is a mesh-free Lagrangian technique used for computational fluid dynamics, particularly in the film industry. The particles represent fluid particles and apply pressure and viscous forces to their near neighbours. Similar methods can also be used for solid mechanics and Silling [11] has coined the term peridynamics for the technique. However a soap film is more a fluid than a solid and so our treatment is more like that of smoothed particle hydrodynamics in that links between particles are continuously being formed and broken.

Consider a system of fluid particles in surface. The mass of the i^{th} particle is m_i and its position is defined by the position vector $\mathbf{r}_i$ which is a function of time. Let us suppose that there is a tension in the 'link' joining the i^{th} and j^{th} particles equal to

$$T_{ij} = \frac{C_{ij} m_i m_j}{a^3} f\left(\frac{r_{ij}}{a}\right) \tag{52}$$

where

$$r_{ij} = |\mathbf{r}_i - \mathbf{r}_j| \tag{53}$$

and C_{ij} is a quantity with units force times length cubed over mass squared. a is a constant with units of length and $f\,(r_{ij}/a)$ is some function which decreases with the separation r_{ij} to such an extent that only neighbouring particles interact.

Consider a uniform virtual membrane strain ϵ (not to be confused with the permutation pseudotensor). The virtual work associated with the i^{th} particle is

$$\frac{\epsilon}{2}\sum_j (T_{ij} r_{ij}) \tag{54}$$

in which the factor of $\frac{1}{2}$ is there because each link is shared by two particles.

If the surface tension in the surface to be modelled is σ, the virtual work per unit area is $2\sigma\epsilon$ in which the 2 is there because the area strain is 2ϵ.

Thus the virtual work per unit mass is

$$\frac{\epsilon}{2m_i}\sum_j (T_{ij} r_{ij}) = \frac{2\sigma\epsilon}{\mu} \tag{55}$$

in which μ is the mass per unit area.

Thus

$$\frac{\sigma}{\mu} = \frac{1}{4m_i}\sum_j (T_{ij} r_{ij}) = \frac{1}{4a^2}\sum_j \left[C_{ij} m_j r_{ij} f\left(\frac{r_{ij}}{a}\right)\right]. \tag{56}$$

We have large number of particles and therefore we can replace the summation by an integral so that,

$$\frac{\sigma}{\mu^2} = \frac{1}{4a^3}\int Crf\left(\frac{r}{a}\right)\frac{dm}{\mu} = \frac{1}{4a^3}\int\limits_{r=0}^{\infty} Cf\left(\frac{r}{a}\right) 2\pi r^2 dr \tag{57}$$

in which C is the value of C_{ij} in the neighbourhood. If we scale $f\,()$ such that

$$\int\limits_{u=0}^{\infty} f\,(u)\, 2\pi u^2 du = 2, \tag{58}$$

then

$$C = 2\frac{\sigma}{\mu^2}. \tag{59}$$

However σ/μ^2 will usually not be constant. Thus we could, for example write

$$C_{ij} = \frac{\sigma_i}{\mu_i^2} + \frac{\sigma_j}{\mu_j^2}, \tag{60}$$

provided that we know σ as a function of μ.

The mass per unit area associated with the i^{th} particle is

$$\mu_i = \frac{\sum_j \left(m_j F\left(r_{ij}/b\right)\right)}{\int\limits_{r=0}^{\infty} F\left(r/b\right) 2\pi r dr} = \frac{\sum_j \left(\left(m_j/b^2\right) F\left(r_{ij}/b\right)\right)}{\int\limits_{r=0}^{\infty} F\left(r/b\right) 2\pi \left(r/b^2\right) dr} = \frac{\sum_j \left(\left(m_j/b^2\right) F\left(r_{ij}/b\right)\right)}{\int\limits_{u=0}^{\infty} F\left(u\right) 2\pi u du} \tag{61}$$

in which $F\left(r/b\right)$ is a weighting function and b is a constant length. This formula is based upon the idea that the integral of the mass per unit area over area is equal to the total mass.

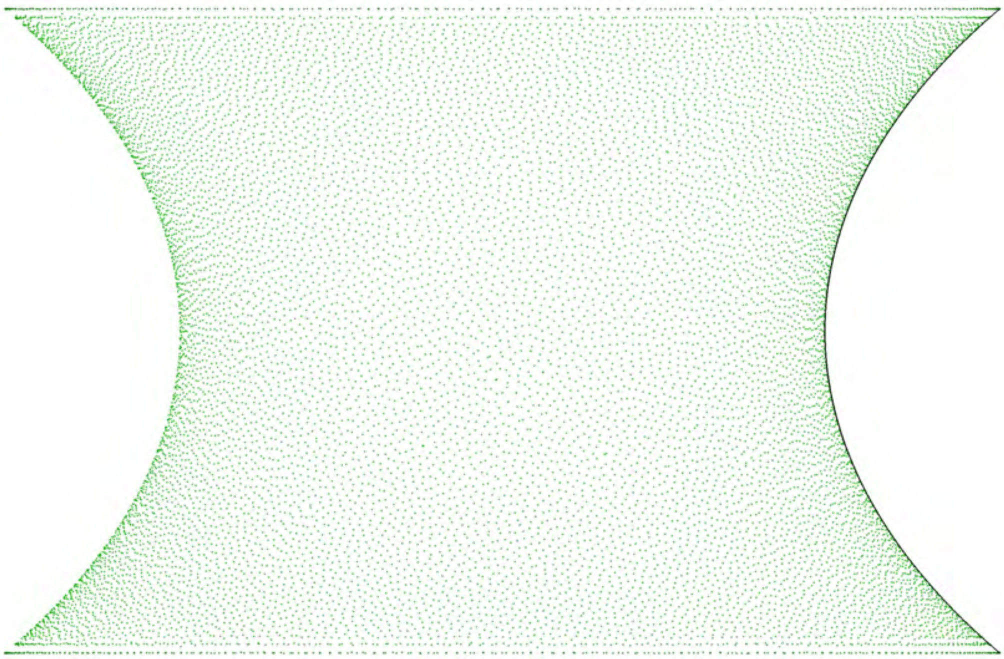


Figure 5: Shell with no load, compared with the catenoid in black

Again it makes sense to scale $F()$ so that

$$\int_{u=0}^{\infty} F(u)\,2\pi u du = 1 \tag{62}$$

in which case

$$\mu_i = \sum_j \left(\frac{m_j}{b^2} F\left(\frac{r_{ij}}{b} \right) \right). \tag{63}$$

It is conventional in smoothed particle hydrodynamics to set

$$f(u) = -F'(u) \tag{64}$$

and this is consistent with our analysis because then

$$\begin{aligned} 1 = \int_{u=0}^{\infty} F(u)\,2\pi u du &= \left[F(u)\,2\pi u^2/2\right]_{u=0}^{\infty} - \int_{u=0}^{\infty} F'(u)\,\pi u^2 du \\ &= -\int_{u=0}^{\infty} F'(u)\,\pi u^2 du = \int_{u=0}^{\infty} f(u)\,\pi u^2 du \end{aligned} \tag{65}$$

The numerical experiments described in the next section used

$$F(u) = \frac{e^{-u^2}}{\pi} \tag{66}$$

and

$$\int_{u=0}^{\infty} F(u)\,2\pi u du = \int_{u=0}^{\infty} e^{-u^2} 2u du = -\left[e^{-u^2}\right]_{u=0}^{\infty} = 1 \tag{67}$$

as required.

We also have

$$f(u) = \frac{2ue^{-u^2}}{\pi} \tag{68}$$

and therefore the force between adjacent particles is zero if $r_{ij} = 0$.

4. Results: numerical particle examples

Figure 5 shows a shell with no load compared with the catenoid in black. The analysis has 27,000 particles and the radius of inflence of each particle is such that each particle interacts with approximately 12 neighbours. Figure 6 is a loaded shell with constant physical stress so that σ/μ^2 in equation (60) is proportional to $1/\mu$. In this case there are 75,000 particles and each particle interacts with approximately 25 neighbours.

In each case only half the shell is drawn so that the particles on only one side are seen. Particles near the line of symmetry are drawn in red in figure 6 to show the shape of the cross-section which should be compared with part of figure 2.

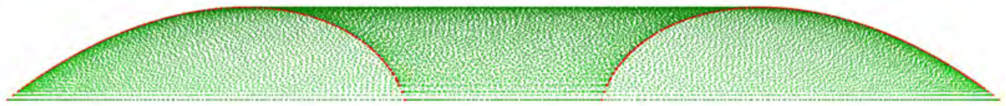


Figure 6: Loaded constant physical stress shell

The shapes were found using Dynamic Relaxation [9] which is essentially the same as Verlet integration [12]. Smoothed particle hydrodynamics is stable because pressure increases with density, but as noted in the introduction a soap film can only be stable if its tension increases as it gets thinner. Numerical experiments were conducted trying to model this behaviour, but with limited success.

It was therefore decided to allow the membrane stress to increase with mass per unit area area - which is what we want for the real shell. This gives us two possibilities:

Tension surface which is unstable within its own plane and stable out of the plane.

Compression surface which is stable within its own plane and unstable out of the plane.

In fact the two approaches are essentially the same. In the stable direction particles are moved in the direction of the out of balance force while in the unstable direction they are moved in the opposite direction to the out of balance force.

The symmetric second order tensor

$$\sum_j \left[m_j \left(\mathbf{r}_i - \mathbf{r}_j\right) \left(\mathbf{r}_i - \mathbf{r}_j\right) \left(\frac{r_{ij}}{a}\right)^2 f\left(\frac{r_{ij}}{a}\right) \right] \tag{69}$$

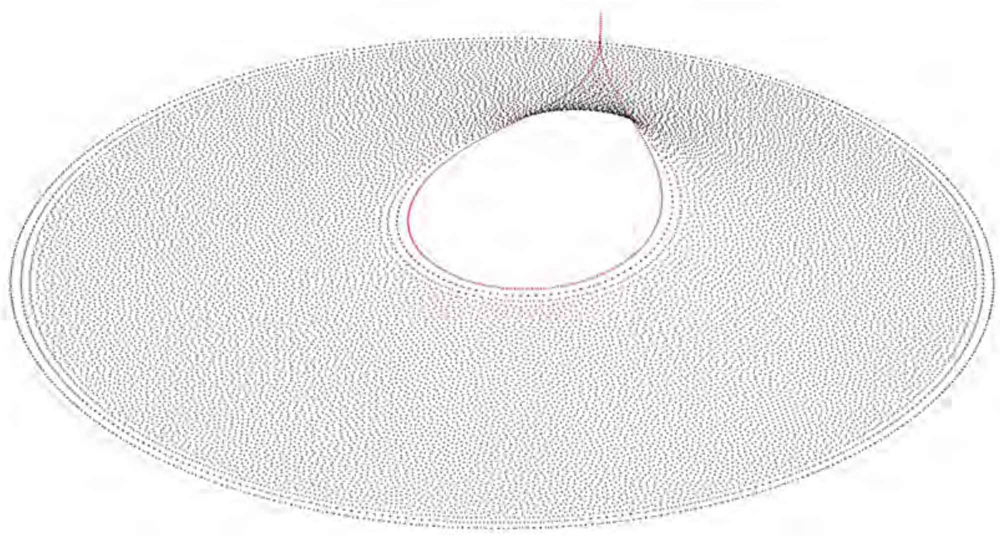


Figure 7: Frei Otto eye

was evaluated at each particle in each cycle. Two of the principal directions of this tensor will lie in the plane of the surface and the third principal direction will be normal to the surface. The principal value associated with the normal direction will be zero and therefore we have a method for isolating the normal component of force.

The reason for the $(r_{ij}/a)^2$ in this expression is to give more weight to particles at a greater distance and so reducing the effect of surface 'roughness'.

The boundary conditions were very simple. The catenoid started as a circular cylinder contained within a cylindrical 'can' with ends. Any particle which moves outside the can is reversed in direction so that there is a concentration of particles where the lid and bottom meet the wall. This concentration causes the layering at the top and bottom.

The boundary conditions are similar for the loaded shell, except for the addition of a 'ball' centred at the middle of the bottom. The initial shape was a parabola rotated around the axis.

Figure 7 is an unloaded soap film supported by a Frei Otto eye, a loop of cotton attached to a fixed point. The outer boundary is the bottom of a 'can'. In this case each particle interacts with approximately 10 particles and there are a total of 19,000 particles. The cotton is simply modelled as

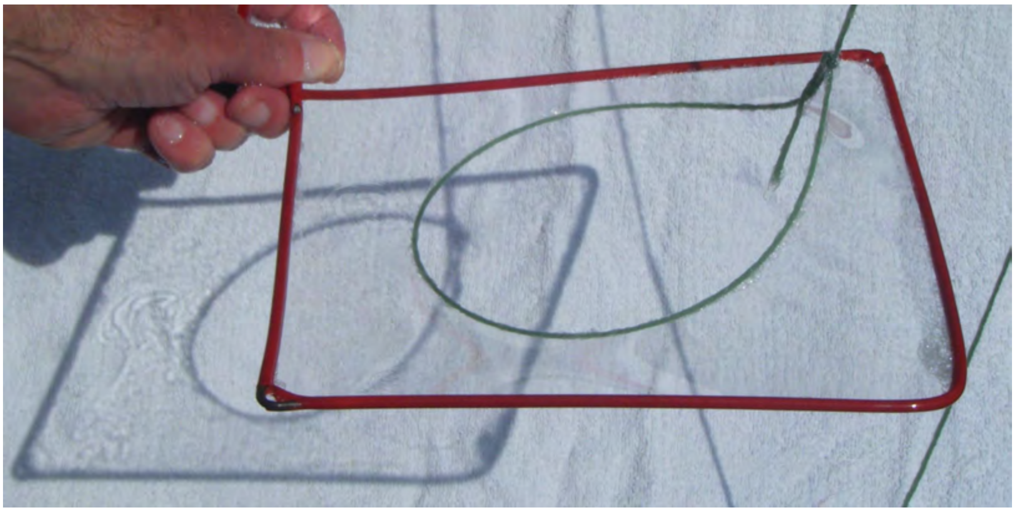


Figure 8: Frei Otto eye - physical model

fluid particles linked by short lengths of cotton. This can be compared with the physical model in figure 8 and with figure 9 which shows the principal curvature net on the minimal surface using the method described in [13]. The tension in the cotton means that its curvature vector must lie in the local plane of the surface and therefore it must be in an asymptotic direction on the surface. Since there is no shear stress in the surface, the tension in the cotton is constant and therefore the magnitude of its curvature, that is the geodesic curvature, must be constant. The outer boundary of the soap film in figure 9 is contained within a sphere within which the soap film is free to slide. This means that the soap film is normal to the sphere and therefore there can be no twist along the edge so that the outer boundary is a principal curvature direction.

5. Conclusions and further work

The numerical work could have been done using traditional finite element techniques which would have been easier and more accurate. But particle methods are interesting because they are more like ‘real experiments’. It should be possible to model a soap film without introducing the reversal of movement associated with negative stiffness which implies that particles have an ‘intelligence’ not associated with real molecules.

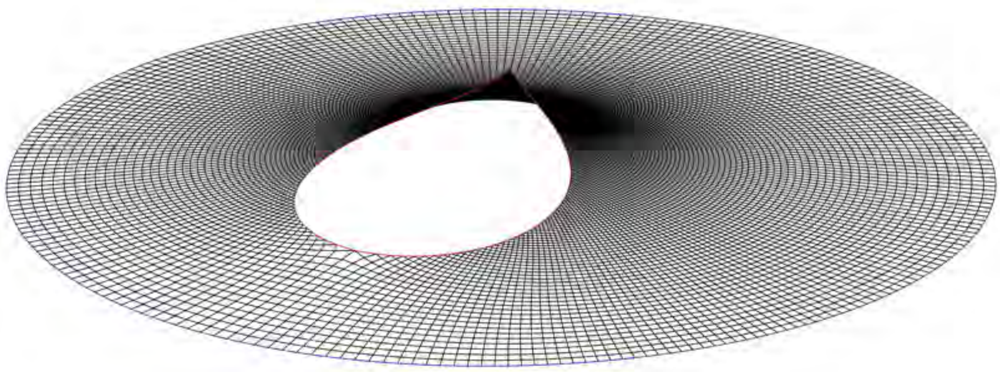


Figure 9: Frei Otto eye - principal curvature net

The technique used for the Otto eye can also be used for the boundaries of a loaded shell structure which can then be inverted to form compression arches.

Having used particle methods for the form finding of a shell, the inter-particle links can be 'frozen' and given elastic stiffness, both axially and in bending. This would allow analysis under load.

References

[1] Williams CJK. The generation of a class of structural forms for vaults and sails. The Structural Engineer 1990;68(12):231–5.

[2] Kilian A, Ochsendorf J. Particle-spring systems for structural form finding. Journal of the International Association for Shell and Spatial Structures 2005;46(2):77–84.

[3] Block P. Thrust network analysis: Exploring three dimensional equilibrium. Ph.D. thesis; Massachusetts Institute of Technology; 2009.

[4] Vouga E, Höbinger M, Wallner J, Pottmann H. Design of self-supporting surfaces. ACM Trans Graphics 2012;Proc. SIGGRAPH.

[5] Gibbs JW. On the equilibrium of heterogeneous substances. Transactions of the Connecticut Academy of Arts and Sciences 1875 to 1878;.

[6] Green AE, Zerna W. Theoretical elasticity. Oxford: Oxford University Press; 2 ed.; 1968.

[7] Timoshenko SP, Gere JM. Theory of Elastic Stability. New York: McGraw Hill; 2 ed.; 1961.

[8] Wright DT. Membrane forces and buckling in reticulated shells. ASCE, Journal of the Structural Division 1965;91(1):173–202.

[9] Barnes MR. Form finding and analysis of tension structures by dynamic relaxation. International Journal of Space Structures 1999;14(2):89–104.

[10] Gingold R, Monaghan J. Smoothed particle hydrodynamics: theory and application to non-spherical stars. Monthly Notices of the Royal Astronomical Society 1977;181:375–89.

[11] Silling S. Reformulation of elasticity theory for discontinuities and long-range forces. Journal of the Mechanics and Physics of Solids 2000;48(1):175–209.

[12] Verlet L. Computer “experiments” on classical fluids. i. thermodynamical properties of lennard-jones molecules. Phys Rev 1967;159:98–103.

[13] Williams CJK. Patterns on a surface: The reconciliation of the circle and the square. Nexus Network Journal 2011;13(2):281–95.

Thinking Topologically at Early Stage Parametric Design

John Harding

University of Bath & Ramboll UK

Sam Joyce

University of Bath

Paul Shepherd

University of Bath

Chris Williams

University of Bath

Abstract. *Parametric modelling tools have allowed architects and engineers to explore complex geometries with relative ease at the early stage of the design process. Building designs are commonly created by authoring a visual graph representation that generates building geometry in model space. Once a graph is constructed, design exploration can occur by adjusting metric sliders either manually or automatically using optimization algorithms in combination with multi-objective performance criteria. In addition, qualitative aspects such as visual and social concerns may be included in the search process. The authors propose that whilst this way of working has many benefits if the building type is already known, the inflexibility of the graph representation and its top-down method of generation are not well suited to the conceptual design stage where the search space is large and constraints and objectives are often poorly defined. In response, this paper suggests possible ways of liberating parametric modelling tools by allowing changes in the graph topology to occur as well as the metric parameters during building design and optimisation.*

1 Introduction

Parametric modelling is now well established within the computational design community. Software applications such as Grasshopper by McNeel & Associates, Bentley Generative Components (GC) and more recently DesignScript by Autodesk allow complex ideas to be explored at the early stage of design that go beyond what is possible using the traditional methods of hand sketching and model making alone. In addition to the generic software platforms, in recent years many third-party analysis plug-ins have also been developed that provide real-time performance feedback to assist at the early stage of design [Shea et al. 2003].

A combination of parametric modelling, performance analysis tools and heuristics allow for a variety of design options to be explored both quantitatively and qualitatively by adjusting numeric parameters. As the most impactful decisions in the design process are made at the start of any project, tools which assist good decision making at this early stage are of great help to the design team.

1.1 One user - one graph - one model

Parametric modelling is not an easy task. As Woodbury and Aish [2005] state: "Designers must model not only the artifact being designed, but a conceptual structure that guides variation. At the same time they must attend to the multi-faceted design task at hand." The recent rise of parametric modelling tools has further emphasised that the process of structuring of the graph has become integral to the design process itself, leading to the term 'parametric design'.

Parametric design requires the user to construct a single directed acyclic graph (DAG) made up of parameters and components. This DAG representation has an ordering of vertices, a so-called 'topological ordering' which is computed in order to generate geometry. This ordering also to some extent expresses the history of the model's creation explicitly, an associative record of how the building geometry is constructed from a series of base level parameters and components.

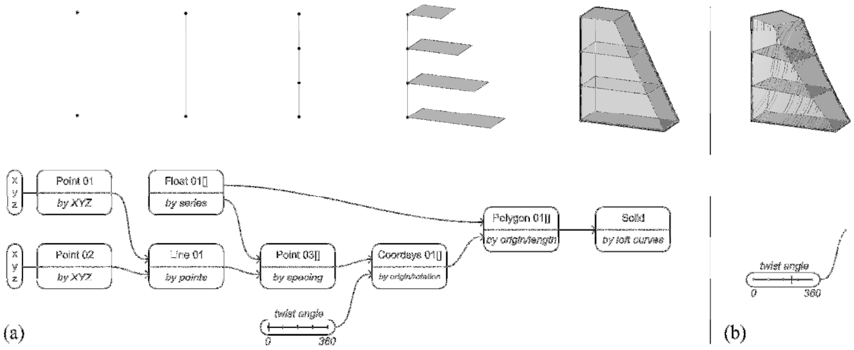


Figure 1: (a) Creation of a simple building form using a directed acyclic graph. (b) Manually moving the 'twist angle' slider enables new design options to be explored within a user-defined range of numerical values.

The increase in initial effort in generating a parametric model compared to a traditional CAD one is paid back later in the design process when various design options can be explored by adjusting parameter values. What these parameters 'mean' is represented in the graph structure, of which the designer has top-down control. The user must have comprehension of the graph itself and the memory of how it was constructed in order to meaningfully invoke alterations to its structure as the design process unfolds. This enforces a limit on the complexity of the graph as it must be understood by at least one human mind. As the parametric graph structure becomes more and more complex, so reduces its flexibility and potential to adapt to changing constraints and requirements.

This single user authorship can be a problem when using parametric design in a collaborative environment, and has been criticised by Aish [2000] and Holzer [2010] for early stage design. In practice, the single user is often the architect with an initial concept created using a parametric model. When passed on to other team members, the graph can resemble a tangle of spaghetti, making it hard to follow geometric relationships [Davis et al. 2011a]. Alterations are therefore limited to adjusting the metric sliders whose relationships have already been defined.

1.2 Combining modelling with analysis & optimisation tools

Adjusting the parameters in any parametric model enables exploration of different design options, each of which can be evaluated by quantitative and qualitative criteria. When used in combination with a multi-objective optimisation algorithm [Deb 2001], multiple designs can be generated and evaluated automatically within the set parameter constraints, with high scoring designs identified and stored.

Current parametric modelling tools are beginning to include such generic solvers as standard allowing bi-directional graph associations [Rutten 2010], [Coenders 2011], hence their importance for architectural design problems is already growing [Evins et al. 2012] and is likely to increase in the future. Initialising such a process assumes that the performance metrics can be expressed quantitatively, however the important addition of subjective judgments during the search is often required in order to include qualitative aspects such as social impact, aesthetics, iconography, etc...

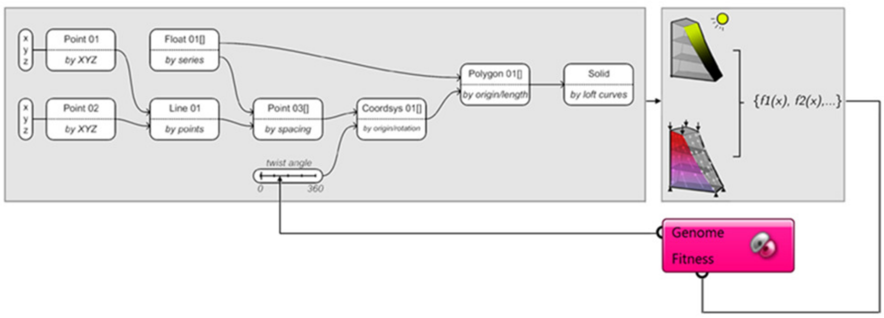


Figure 2: A parametric model is optimised for multi-objective performance criteria. In this example, the limits of a single slider provide the search space domain.

At present such methods require that the associations between the elements in the graph, or 'body plan' [DeLanda 2002] remain constant; it is only the metric values that can be made variables. This usually means that only one building typology can be explored per parametric model. While this is satisfactory should the building type be already agreed upon, if parametric design tools are to be increasingly employed at the conceptual design stage such a lock-in of the graph structure is not conducive to the exploration of multiple building configurations.

As an example, in a recent collaboration with Bjarke Ingels Group architects, one of the authors was required to give quantitative performance feedback on over one hundred tower design options at the concept design stage (Figure 3).

Figure 3: Foam scale model design options for a tower project at concept design stage.

For each design, an evaluation of relative structural performance was required in order to better decide which direction the design process should take. This evaluation could not be derived from the foam scale models alone and so a computer modelling process was required. However, after testing many of the current CAD and parametric design tools it became clear that no package was available that could adequately generate models for each design typology within a short period of time. In effect, a completely new parametric model needed to be built to adequately represent each design option – something that was impractical at the concept design stage. In the end, the number of options had to be narrowed down significantly before any structural analysis could be undertaken meaning that potentially good design directions were missed. In addition to metric variations, had we also been able to think topologically and automatically generate different graphs representing different building types this may not have been the case.

2 Top-down graph making

The relationship between the graph representation and the geometric model is many-to-one, that is, we can create two graph structures that both produce an identical geometric model. A simple example is shown in Figure 4 where a different graph generates exactly the same geometry as in our earlier example (Figure 1) but in a different way.

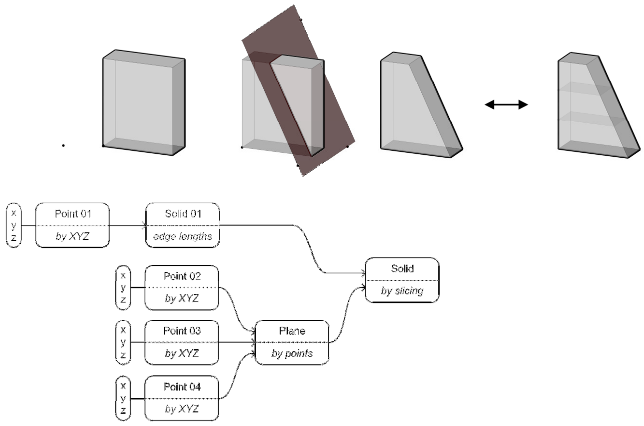


Figure 4: An alternative graph leading to the same tower design as shown in Figure 1.

When comparing each method used to generating the final form, it may seem tempting to use Occam's Razor and prefer the simplest graph representation, however one of the strong benefits of using a parametric model is that it can explore a range of possible designs. For that reason it is sometimes the graph with the highest amount of variability that may allow the greatest freedom of exploration. Different graph structures make explicit different design intentions or investigations. This means that although the initial form may be identical, how it is represented in the graph will influence its future development. This is shown in Figure 5, whereby adjusting the parameters for each graph enables exploration of completely different solution domains and hence building typologies.

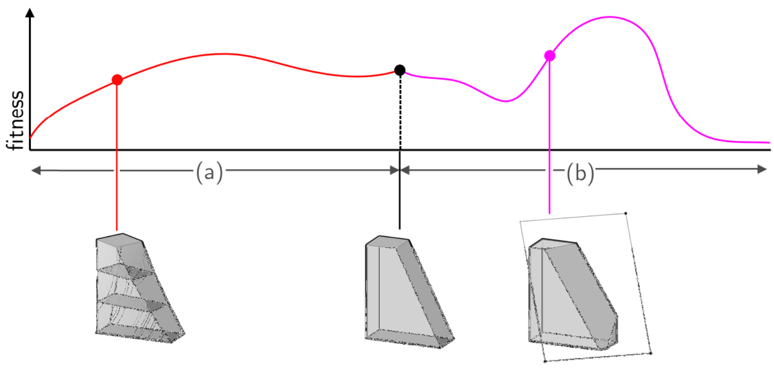


Figure 5: From the same initial tower model geometry, a different graph representation leads to exploration of different design domains (a) & (b).

By using the one-graph, one-model approach that parametric design requires, the user must choose how to set up a parametric model early in the process. However, design requirements and performance evaluation criteria are likely to be subject to change as the design process develops. In our example, as we progress along the design path and our freedom becomes ever more constrained, it may be that the design converges towards a sub-optimal solution (a) instead of the better option (b), even with the help of parameter optimisation algorithms as described in Section 1.2. Worse still, there is no way of knowing that the design is sub-optimal because we have only explored the domain specific to one graph structure, thus creating the illusion of an 'optimal design'.

One option is to alter the graph manually, either to widen exploration or to change design direction. In practice this can become a problem because as the complexity of the model increases the dependencies become more difficult to adjust and design freedom actually decreases. As a result, the initial parametric relationships tend to get 'locked-in' and cannot adapt. This experience was found by Holzer et al. [2008] on a stadium roof project "whereby changes required by the design team were of such a disruptive nature that the parametric model schema could not cope with them". As Aish and Woodbury [2005] state: "It is crucial to recognise that nothing can be created in a parametric system for which a designer has not explicitly externalised the relevant conceptual and constructive structure. This runs counter to the often-deliberate cultivation of ambiguity that appears to be part of the healthy design process."

2.1 Faster graph manipulation methods

One response to this issue is to make the graph representation easier to change manually and hence more flexible. Such methods include reusing commonly found design patterns [Woodbury et al. 2007], better structuring using principles from modular programming [Davis et al. 2011a] and the combination of user-created undirected graphs with a logical interpreter [Davis et al. 2011b]. Conventional strategies of good source code management and documentation can also help as well as innovations such as the transactions concept in GC, whereby discrete pieces of logic are recorded at key stages of graph development. Such methods offer improvements to both the speed of graph manipulation and their legibility as collaborative models; however they are all still based around wilful modifications being made to a single graph topology that must be explicit and intelligible. The fundamental digestion of a graph's associative complexity is still problematic.

In response to the problems encountered when humans attempt top-down control of the parametric graph, instead might it be possible to think topologically and automate the generation of the graph itself, opening up exploration of different building typologies as required at the concept design stage?

3 Meta-parametric modelling

The modifications in Section 2.1 suggest ways to improve the flexibility of parametric models. However, it is our opinion that whilst the creation of the parametric graph structures remains predominantly under top-down control, the vast search space at conceptual design stage cannot be properly explored. By thinking at a higher level of abstraction, the authors propose automating the process of graph generation itself alongside variation of the metric parameters. Such an approach is similar to genetic programming whereby the automatic generation tree structures [Koza 1992] and even directed acyclic graphs [Van Leeuwen 1990] takes place which represent computer instructions.

Automatic generation of parametric graph structures would potentially enable different building typologies to be explored, even if the variation of the graph structures be fairly minimal. Figure 6 shows how thinking topologically with our parametric and optimisation design tools was exactly what was required on the tower project discussed in Section 1.2.

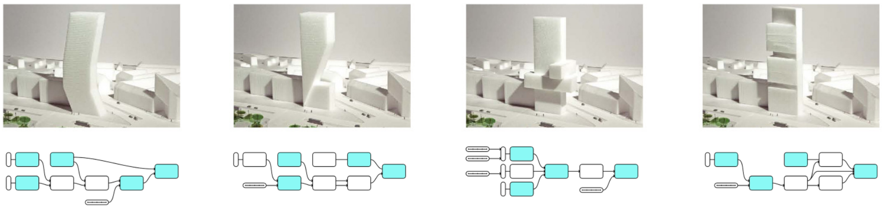


Figure 6: Simple tower block models are represented by different slider metric values and graph structures.

3.1 Permutations

One can explore the combinatorial possibilities associated with constructing an automatic generator of valid parametric models. Figure 7 shows a range of permutations generated from a stepwise addition of nodes based on three cases (two types of nodes or both) which can only be applied to the newest node. Each time a node is added a different parametric model is produced, it is possible to imagine generating all possible models by iterating through all valid permutations that the parametric graph concept permits, although some graphs may produce the same design as others the size of all the possible combinations is still huge. This is evident even if only a limited number of available nodes are considered for a low graph complexity as shown in our example.

It is clear that by thinking topologically as well as numerically in any computer search, the permutations will increase massively. However, the amount of graph variation allowed could still be under the control of the design team to some extent. Even if the structure of the graph could explore just two or three different building types it would still be an improvement on the current situation of one concept design per parametric model.

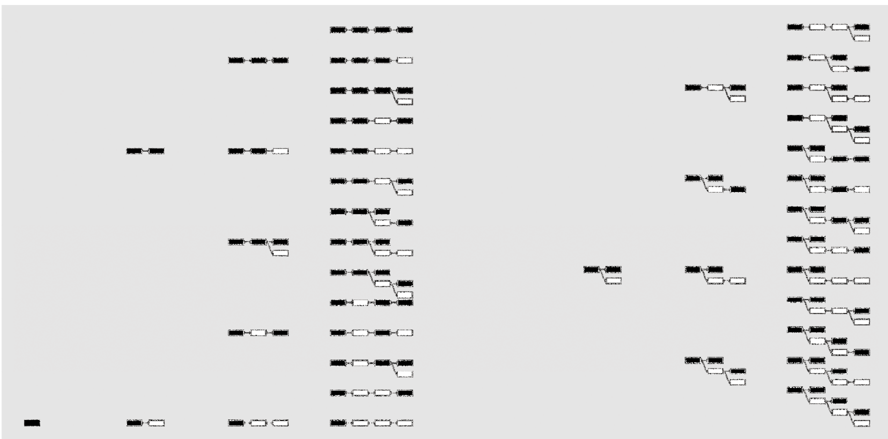


Figure 7: A search through some possible permutations of a DAG with the 'seed' design at the bottom left of the picture.

3.2 Adaptation

Architectural design is a 'wicked' problem that rarely provides concrete problems with fixed constraints and goals [Rowe 1986]. This is especially the case when modelling designs at the early stage, as Mueller [2011] states: “As the model is often constructed at the concept design stage, it is unrealistic to assume that a definitive, consistent set of design requirements can always be established ahead of time in order to provide an ironclad test for a solution.”

As well as broadening the search space, moving away from a fixed parametric graph schema also means a design model could potentially adapt to future changes in constraints and requirements much better and avoid becoming ‘locked-in’ as discussed in Section 2. For example, should the client change the gross floor area during design development, a different building type maybe better suited to the task when considered in the context of other requirements. As discussed earlier, a regular parametric model is unlikely to be as easily adaptable to such requirement changes.

3.3 Authorship

Parametric modelling has always involved top-down control in generating the graph topology, choosing its parameters and components, and hence the building typology. This one-model approach can be extended to allow multiple users to add their own input [Hudson et al. 2011] so long as the graph is intelligible, however such examples involve only a single graph and associated building type in order to establish common ground between stakeholders. This dependence on a single design model is similar to the Building Information Modelling (BIM) philosophy.

With a multiple graph generator, the components used would still need to be specified as well as their combinatorial rules (i.e. line by points, surface by loft curves, etc...), however how they are associated in a particular graph is now open to the optimisation process. The possibility of creating graphs that are incomprehensible for any human mind would also be possible, with the associated building geometry still able to be evaluated both quantitatively and qualitatively as a ‘phenotype.’ Simple low-level rules leading to highly complex graphs.

Finally, there could be interesting consequences in automating graph generation in terms of project workflow. It could potentially lead to more effective collaboration in a design team, as no single stakeholder can lay claim to have overall authorship of the model as is the current practice. Furthermore, this method obfuscates the process of model creation but in doing so emphasises the need for users to develop a better joint understanding of the various building centric performance requirements, their relative importance and how best to set the low level combinatorial rules that generate vastly different building models.

4 Conclusion

In this paper we have highlighted that current parametric modelling tools emphasise the one-user, one-graph, one-model approach to design with any search process conducted by the computer limited to metric parameters only. Whilst such a limiting approach can be highly effective when the proposed building type is known, we argue that at the conceptual design stage we must instead think topologically in order to facilitate a wider design exploration.

We have proposed that the graph generation itself instead of being limited by top-down creation should be open for change in any search process and hence in effect be generated bottom-up. This involves moving to a higher level of abstraction when thinking about parametric design. This method undoubtedly has many issues associated with its practical implementation, which can only be discovered in further development and realisation of these ideas. However, we believe that this approach could potentially offer a new way of approaching early stage parametric design which deserves further exploration. It is hoped that this paper will inspire others in the field to move towards thinking not just numerically but topologically in parametric design and optimisation.

Acknowledgements

This work has been supported by Ramboll UK and the EPSRC funded Industrial Doctorate Centre in Systems at The University of Bath (Grant EP/G037353/1). Our thanks go to Cat Huang at Bjarke Ingels Group for granting image permission for Figure 3.

References

AISH, R. 2000. Migration from an individual to an enterprise computing model and its implications for AEC Research. *Berkeley-Stanford CE&M Workshop.*

AISH, R. AND WOODBURY, R. 2005. Multi-Level Interaction in Parametric Design. In *Lecture Notes in Computer Science 3638*, 151–162.

COENDERS, J.L. 2011. NetworkedDesign, Next Generation Infrastructure for Design Modelling. In *Proceedings of the Design Modelling Symposium 2011*, Springer-Verlag Berlin Heidelberg, Berlin, Germany, 39-46.

DAVIS, D., BURRY, M., AND BURRY, J. 2011a. Untangling Parametric Schemata: Enhancing Collaboration through Modular Programming. In *CAAD Futures 2011, Designing Together*, 55-68.

DAVIS, D., BURRY, M., AND BURRY, J. 2011b. The flexibility of logic programming: Parametrically regenerating the Sagrada Família. In *Proceedings of the 16th International Conference on Computer Aided Architectural Design Research in Asia*, 29-38.

DEB, K. 2001. *Multi-Objective Optimization using Evolutionary Algorithms.* John Wiley & Sons.

DELANDA, M. 2002. Deleuze and the use of the genetic algorithm in architecture. In *Designing for a Digital World*, 117–120.

EVINS, R., JOYCE, S., POINTER, P., SHARMA, S., VAIDYANATHAN, R., WILLIAMS, C. 2012. Multi-objective design optimisation: getting more for less. *Proceedings of the Institution of Civil Engineers Civil engineering Special issue* 165(5), 5-10.

HOLZER, D. 2010. Optioneering in Collaborative Design Practice. In *International Journal of Architectural Computing*, 8(2), 165-182.

HOLZER, D., HOUGH, R. AND BURRY, M. 2008. Parametric Design & Optimisation for Early Design Exploration. In *International Journal of Architectural Computing*, 04(05), 638.

HUDSON, R., SHEPHERD, P. AND HINES, D. 2011. Aviva Stadium: A case study in integrated parametric design. In *International Journal of Architectural Computing*, 9(2), 187-202.

KOZA, J.R. 1992. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, MIT Press.

MUELLER, V. 2011. Distributed Perspectives for Intelligent Conceptual Design. In *Distributed Intelligence in Design*, Wiley-Blackwell, Oxford, UK.

ROWE, P. 1987. *Design Thinking.* MIT Press.

Rutten, D. 2010. Evolutionary Principles applied to Problem Solving <*http://www.grasshopper3d.com/profiles/blogs/evolutionary-principles*> accessed 14th May 2012.

Shea, K., Aish, R. and Gourtovaial, M. 2003. Towards performance based generative design tools. In *Digital Design, 21st eCAADe Conference proceedings*, 103-110.

Van Leeuwen, J. 1990. Graph algorithms. In *Handbook of Theoretical Computer Science Volume A: Algorithms & Complexity*, Chapter 10, Elsevier, Amsterdam.

Woodbury, R., Aish, R. and Kilian, 2007. Some Patterns for Parametric Modeling. In *Proceedings of 27th ACADIA Conference, Association for Computer Aided Design in Architecture*, Halifax.

Linear folded V-shaped stripes

Rupert Maleczek,

University Innsbruck, Institute of Design / unit structure and Design -koge, Austria

Chloe Genevaux,

Laboratoire Géometrie Structure Architecture, Ecole Nationale Supérieure d'Architecture de Paris-Malaquais, France,

Sam Conrad Joyce,
University of Bath, United Kingdom

Abstract

This paper presents research to find a computational method for creating freeform structures consisting of simple linear folded V - shaped stripes.
A geometric algorithm produces a series of stripes that form regular and irregular reticular structures on a given surface (Fig. 1). This algorithm enables the approximation of single to double curved surfaces. The V- section form of the stripe has advantages over other known folded stripe systems by adding rigidity to the stripes and whole structure. Indeed, simple linear folded stripes can be considered as half reverse folds. Being rectangular in unrolled condition, the stripes undergo no torsion when folded.
This system can be classified as a post defined open stripe system (Maleczek, Geneveaux 2011).

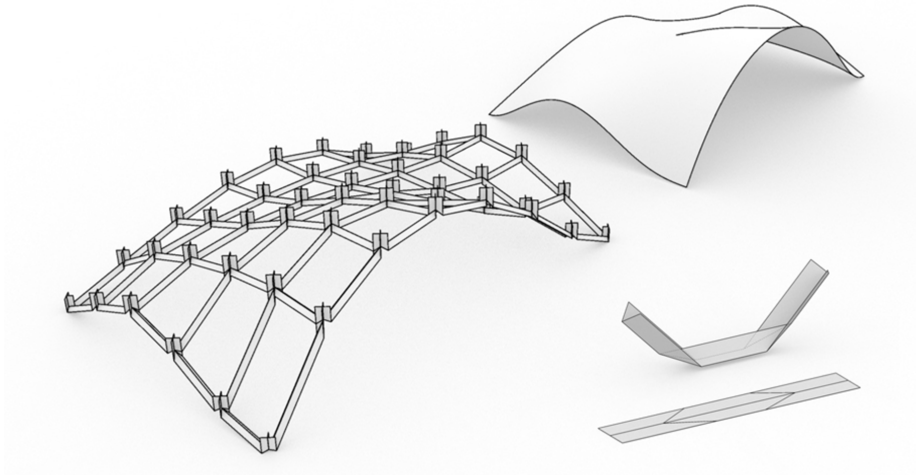


Figure 1: a surface structuralised with L-shaped stripes, and an example of its genotype

V-SHAPED STRIPES

The genotype of this technique is a popular and often used element in these types of structures. It has been described as reverse folded stripes (Buri 2010), as rigid isometric origami (Klett; Drechsler 2011), or as unit with isotropic vertices (Tachi 2009). In its unrolled position it forms a rectangular stripe, with three folds in the length along the middle axis, and two additional folds on each side (Fig. 2).

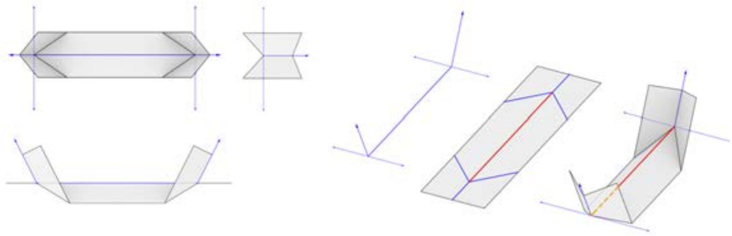


Figure 2: a regular V-shaped stripe

The centred fold along the middle axis is a mountain fold, while all other folds are valley folds. The angle between the side folds and the middle fold has to have a variation of 90 degrees; otherwise the stripe can not be folded. More than two side folds can be formed on the same stripe. This folded stripe has one degree of freedom that enables the variation of dihedral angles between the stripe's planes. In a stripe, all dihedral angles formed by the planes separated by the middle fold are equal, only their direction is inverted.
This folding technique is usually used in a surface approach, by multiplying the number of mountain and valley folds on a unique corrugated surface. The authors are interested here in the assembly of V-shaped stripes into reticular structures, using star-like nodes. This strategy can be seen as an alternative to an approach where the structure consists of large folded panels, instead providing a reticular structure that consists of relatively small folded elements.

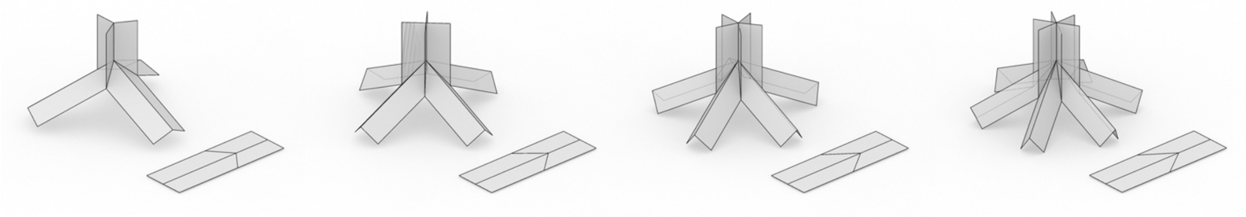


Figure 3: regular star-nodes with 3,4,5 and 6 elements

The number of stripes that can be assembled in a star-like node is variable (Fig3). To have a V-shaped section of the base module, a minimum of three elements must be assembled. In this paper; regular nodes, semi-regular nodes and irregular nodes will be distinguished. A regular node is defined by equal angles between the middle folds, thus all dihedral angles are also equal. In regular nodes the number of assembled elements directly defines the dihedral angle. Nodes with various angles lead to different dihedral angles, and can be described as irregular nodes.
In this paper different techniques and algorithms for surface approximation will be presented. Curved surfaces will be approximated by polygonal faces. The advantages and limits of this approach will be presented from a geometrical as well as from a structural point of view.

Stripe-Elements and their Generation

V-shaped stripes can be defined as a variation of simple linear folded stripes. A simple folded stripe can be considered as a half of a V-shaped stripe, cut along the middle fold. The main difference is the assembly technique. The faces of each stripe can be defined as connection- and contact-segments. While contact segments are connected together to manage the assembly of stripes, connection segments connect two contact segments, belonging to the same stripe. In this particular system, the connection-segments are joined to each other through a fold along the middle axis of the stripe. To assemble stripes together in a reticular structure, a reverse fold is created at each end of the stripe. In this configuration, each stripe will have a minimum of one middle fold in its length, forming the "Middle-Axis"(Fig. 4) and four side folds. In most cases, each contact segment is connected to a contact segment, which belongs to another stripe. Therefore the number of contact segments of each stripe defines the number of stripes it is connected to. In this paper the majority of the described stripes are connected to four other stripes.

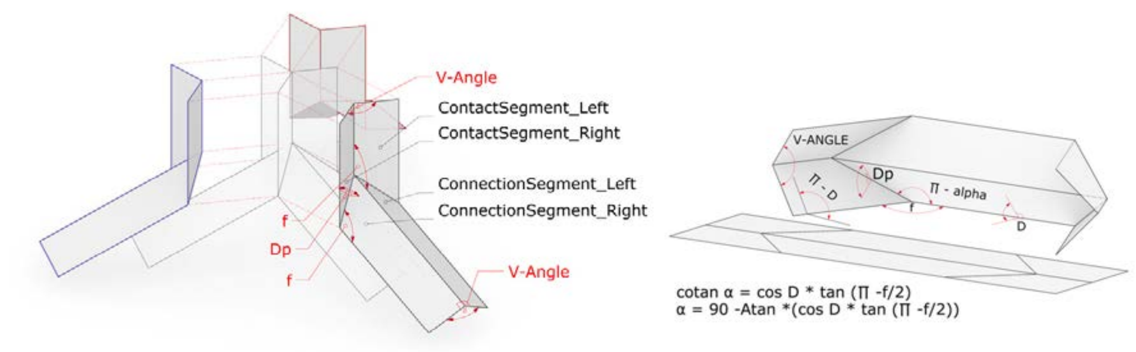


Figure 4: a regular node assembly with the stripe elements and their correlation

As V-shaped stripes are defined by the assembly method, it is beneficial to generate the stripes from points and vectors (Fig. 5). Each stripe can be generated from two points and three vectors connected to each point. For each stripe, the middle axis can be generated between these two points. In each case, two vectors are representing the middle-axis of the neighboring segments, and one vector, the common direction of all folds connecting segments on the star-like node. This direction is identified as the Pin-Direction.

The V-angle is the bisector of the angle between the middle-axis of the neighbouring segments. If all neighbouring middle axis have the same angle relative to its neighbours measured in the pin direction, the star-like node can be described as a regular node. For the regular node the correlation between the different angles can be described as in Fig. 4. If all nodes in a reticular structure are the same regular nodes, this structure can be described as a regular tessellation.
If the stripes are generated from a mesh, then a mesh edge with its neighbouring mesh edges and a direction at its vertices will be required to generate a stripe along each edge.

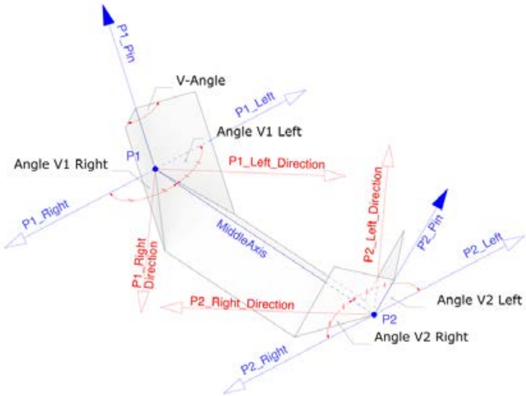


Figure 5: Elements needed for the stripe creation

Regular Star-like Nodes and their corresponding Tessellation

As all dihedral angles in a regular star-like node have to be equal, a grid of regular nodes must fulfil this relation in all nodes. There are three classical regular tessellations, which will be described here (Fig. 6). The dihedral V-Angles can be defined by a complete circle (360°) divided by the number of middle-axis meeting at each node. This establishes triangulated grids with an equal dihedral angle of 60 degrees (360°/6) , rectangular grids with an equal dihedral angle of 90 degrees (360°/4) , and hexagonal grids with an equal dihedral angle of 120 degrees (360°/3). With this formula, the V-angle for each stripe connected in a regular node can be calculated. These angles must be have an equal measurement from the pin of each node.

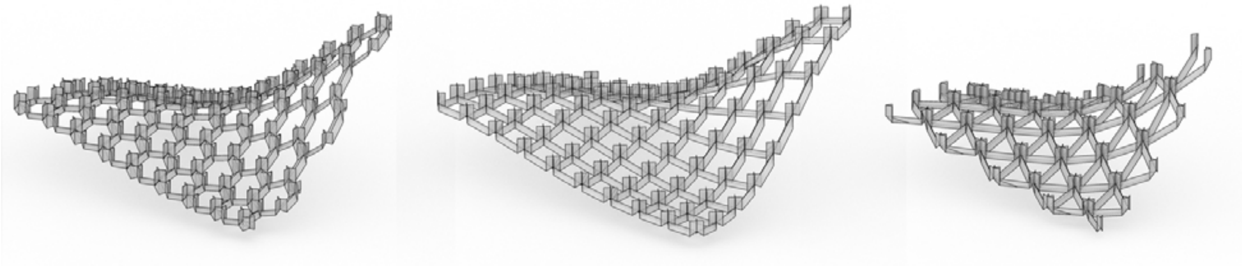


Figure 6: from left to right: a hexagonal, a quadrahedral and a triangulated regular tessellation on the same surface

Therefore it is useful to create regular tessellations with pin directions that intersect either in a point or in infinity. In the special condition, of meeting in infinity, all pins are parallel. The advantage of regular grids with regular star-like nodes lies in the fact that the V-Angles of all stripes are equal.

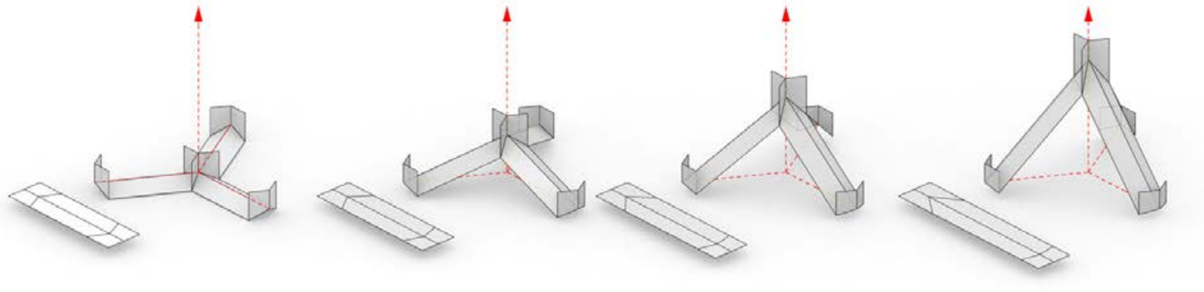


Figure 7: a regular node with changing height, by a constant V-Angle

Regular tessellations allow for a very interesting technique to approximate doubly-curved surfaces. As all V-angles within a regular tessellation are depending on the number of elements in the node, and must be measured in a plane, where the pin direction is equal to the surface normal, the point in the node itself can be moved along the pin direction, with no influence on the V-Angle, but a change in all angles alpha of the folds measured to the middle axis and the folding angle Dp in these folds (Fig. 7), depending on the change of the angle f.

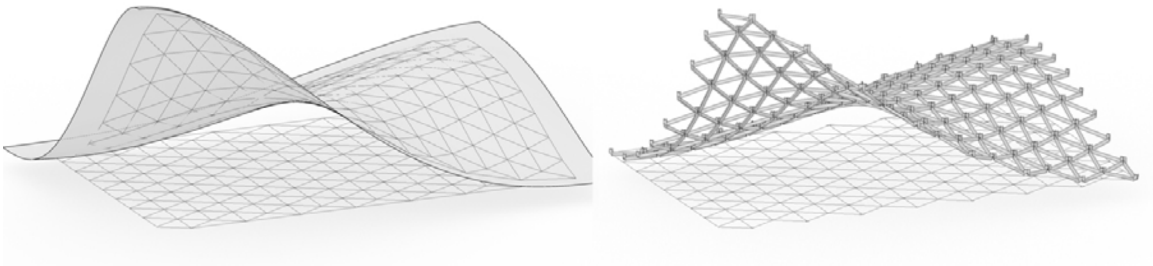


Figure 8: a triangulated regular grid projected to a surface.

One simple strategy to approximate surfaces with regular grids can be a projection of a regular grid of lines to a surface in pin direction (Fig. 8). Projected to a sphere this strategy will lead to an icosahedron, if the grid is triangulated. In this icosahedron not only the V-Angle is equal in all stripes, but also the fold angles Dp are equal. A flat surface structuralised with a regular grid, will generate equal stripes, that have the same V-Angle and equal fold-angles Dp in every fold. As soon as the approximation of a double curved surface is needed, the V-Angle remains the same in all stripes, but the folding angle Dp and the correspondent angle α will be different in most folds of the stripes.
One possible strategy to minimize the number of different folding angles could be placing limitation on the height distance of neighbouring points, to fixed values, so there will only be a limited number of folding angles. With the adjustment of the steps only, without taking into account its neighbours, the number of different angles is reduced enormously. Here the density and the allowed steps will define the number of different angles within the stripe system (Fig 9). One issue here is the decision on the "smoothness" of the resulting structure.

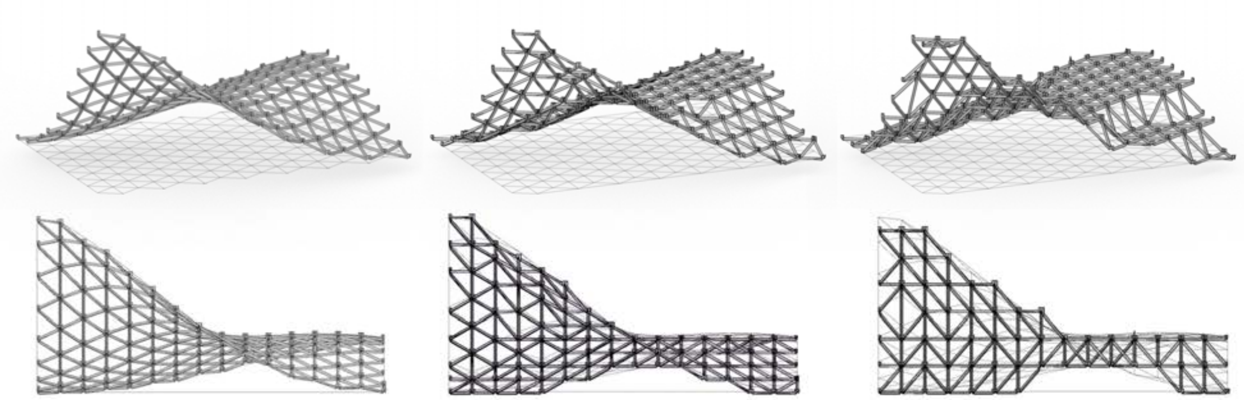


Figure 9: adjustment of the steps in the direction to approximate a surface with a fixed number of different stripes

The method described above, has some limitations concerning the feasibility and approximation of possible surfaces. One limitation is for parallel-stripe systems is the direction of the surface normals in relation to the pin direction (Maelczek 2010). If the angle between the surface normal and the pin direction is larger than 90 degrees, the regular star-like node will no longer work within the system. In other words, surfaces with undercuts will not work with regular tessellations (Fig. 10).

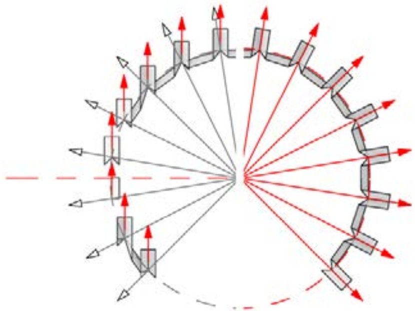


Figure 10: the limit of parallel pin directions (left) and the possible solution with radial pin directions (right)

One possible solution would be the development of a stitching method. In order to create buildable structures, it seems to be beneficial to avoid surface normals directions that get close to this orthogonal condition. Some surfaces allow an approximation with pin directions aligned to a centreline (Fig. 11). Therefore a radial grid around this centreline is projected to the surface. Here the undercut in radial pin direction poses the same problem as in parallel condition.

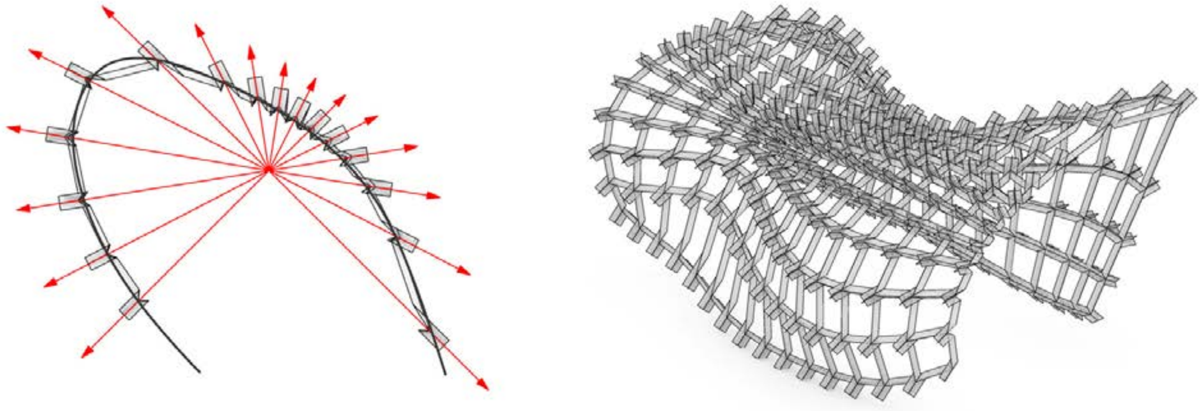


Fig 11: an example for the limit of parallel pin directions and a surface structuralised with radial orientation of the pin directions

Another problem lies in the surrounding border of regular grids. Here in the case of the nodes, either the number of folds will differ, to the regular nodes in the grids, or the number of members is the same, but the angles will differ from the regular star-node.

Genotype-variations and their corresponding tessellations

The Genotype described above works on a wide variety of surfaces but is, as described, limited and restricted for parallel or centred pin-directions in combination with regular equal angled middle-axis. In order to extend the degrees of freedom of this reticular folding system, there are several possibilities and approaches. As one main concern of the paper is to keep the stripe rectangular in its unrolled condition, the authors propose two variations of the genotype. While the first solution works only under very strict boundary conditions, the second one enables a wide variety of forms and possible tessellations by introducing up to two "double-folds"(Maleczek, Genevaux, Ladinig 2012).

Semi-regular star-like-nodes

In this approach, the number of folds within the stripe member will not be changed. Therefore it is necessary that the angle and vector relationship follows very strict rules. The dihedral angles V in both pin directions have to be parallel, and the diagonal facing adjacent directions P1_Left and P2_Right as well as P2_Left and P1_Right have to be parallel. If these three conditions are fulfilled, it is possible to keep the number of folds constant. Therefore, one fold between two contact segments must be rotated. If this solution is chosen, the node can be described as irregular, caused by the fact, that the connected contact faces, will not be connected with the full surface area. In comparison to a stripe based on a regular node, this approach will also rotate the V-Angle around the baseline, if the stripe is creating one regular node, and one semi regular node (Fig 12).

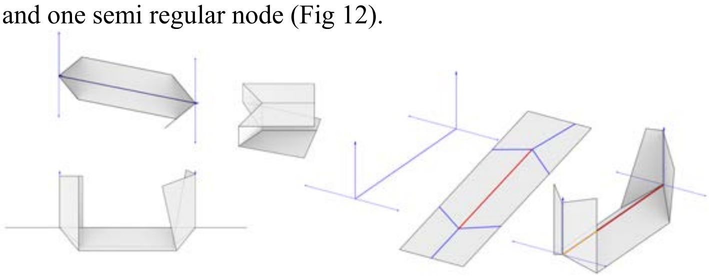


Figure 12: a semi-regular V-shaped stripe

One big advantage of this system is the assembly of semi-regular tessellations (Fig. 13). A semi regular tessellation is a grid, where all nodes are connecting the same number of stripes, and the V-Angle for all members in the structure is the same. The main difference is that the adjacent middle axis of the neighbouring stripes, are no longer defining the V-Angle. In order to keep the V-Angle constant, it must be calculated from fixed directions in all nodes within the tessellation. If the middle axis of a stripe is in-between these fixed directions, this semi-regular V-shaped stripe can be used. Therefore semi-regular grids can be seen as deformed regular tessellations.

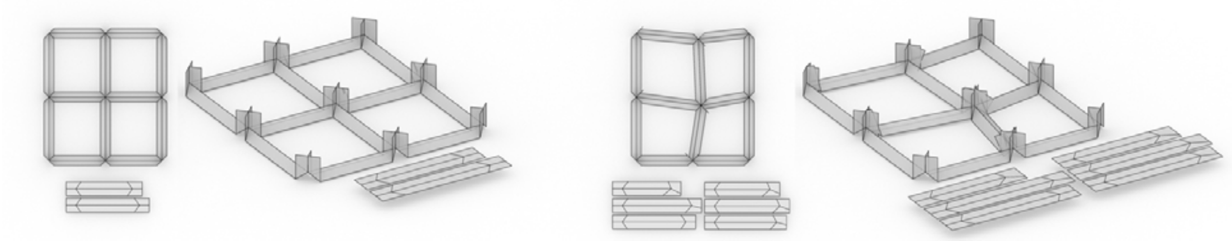


Figure 13: a regular tessellation (left) and a semi regular tessellation (right)

Here the star-like nodes exhibit the same angles as in a regular grid, but the middle-axis must no longer be in the exact centre of the V-Angle. All stripes have one degree of freedom, and are rectangular in unrolled position. In order to approximate surfaces with stripes that have only one degree of freedom, semi-regular grids offer a good solution. As semi-regular tessellations, can be seen as deformed regular tessellations, the reticular structure, can consist of both, regular V-shaped stripes and non-semi-regular V-shape stripes (Fig 14).

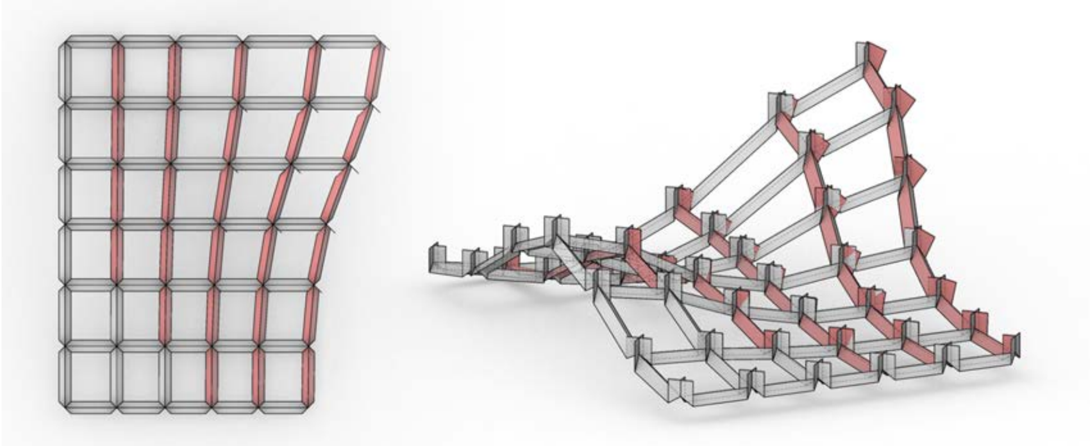


Figure 14: a deformed regular tessellation with the semi regular stripes in red

Irregular Star-like Nodes and their corresponding tessellations

Regular stripe configurations have, per definition, equal pin directions and equal angle relations. If these relationships are no longer fulfilled, the stripes need, in most cases, additional folds. Depending on the relationships of the different angles, one or more double folds are necessary in order to create a linear folded stripe. These double folds, already described for mesh-based stripes (Maleczek,Genevaux,Ladinig 2012), enlarge the possible node configurations, and grids of reticular structures enormously. A double fold in V-shaped stripe allows, for the connection of nodes with differing numbers of members and therefore different angles. As soon as regular or semi regular grids change their pin directions from regular to irregular, this genotype will be very useful (Fig 15).

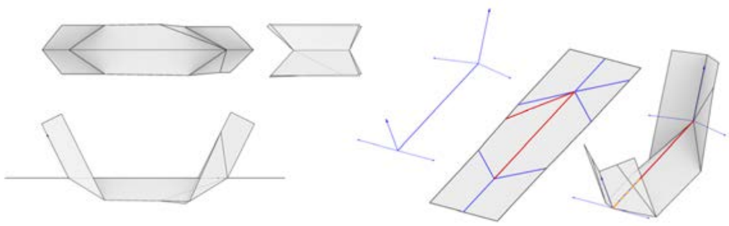


Figure 15: an irregular V-shaped stripe

For architectural use, this method can not only be used to extend tessellation strategies on a given surface, but can also be used to approximate mesh geometries. For a structuralisation of a mesh geometry, the mesh edges will represent the middle axis of the stripe members, and the vertices will represent the nodes. Here, not only triangulated meshes, but also quadrilateral meshes can be used to generate a reticular structure. The pin directions within the structure itself can either be defined through the vertex normal of the mesh or through other constraints. As there is a body of research in the optimisation of freeforms for architectural use, meshes provide a powerful tool to generate stripe based structures from (Sheppard 2011). From form-finding to planar mesh faces a wide variety of solutions are available, for the generation of meshes. These techniques and tools are not only provided as theoretical knowledge but also as tools and add-ons for existing software packages. In other words, the use of mesh geometry for the production of V-shaped stripes opens up a wide field of approaches to generate geometry and form.

The Structural implications of Liner folded V-shaped stripes:

There is much to support the use of the V-shaped stripes. Whilst the benefits of a geometry which is easily constructed out of flat sheets with few liner folds is obvious, the structural consequences and benefits are not so trivial to grasp. In this section some structural implications of the reticulation strategy will be outlined.
Firstly the generation of the structural members in the form of V-shaped structural sections inherently provides for moment resisting depth. While it is generally desirable for such reticulated forms to work as shells with forces travelling in the plane of the surface of the structure, invariably out of plane forces are present which incur bending in the structure that it then must resist. The structural depth afforded by the V fold effectively provides this resistance. There are some impacts on this benefit however brought about by pin direction. The optimal case for structural depth is where the pin directions are parallel with the local surface normals, here we develop the largest out of plane depth, conversely pin angles which approach tangent to the surface have close to zero geometric depth measured normal to the surface and represent low performing structural designs (Fig 10). Thus this gives further incentive to choose pin directions effectively.
Additionally the V shape section is a relatively stable geometry during axial compression, it should also be noted however that if the grid is sufficiently large bucking might become issue. In this case there would be considerable benefit in closing off the V section to provide a more stable section.

The connection detail of the stripe nodes have the advantageous quality of the top folds of the stripes joining to a single point with respect to the stripe centrelines. This allows for simple resolution of forces promoting better shell action of the structure. Looking in more detail the stripes directly bare on to each other allowing for easy fixing strategies. It should be noted that in the case there are multiple edges, which have significantly different angles from the node pin direction to each other then there will be significantly varying depths of the beams. This results in varying second moments of area for stripes made of same width material, and if substantial this could lead to unequal out of plane stiffness and less even and effective distribution of forces.
One concern which would require further investigation is the potential for the structure to generate stress concentrations around joints in the folds at the nodal points, however such investigations are out of the scope of this paper, but could potentially be mitigated by appropriate reinforcement which should be considered for latter study. Due to the relatively high geometrical freedom of the system to place nodal positions, there is much scope to apply relaxation techniques (Williams 2001) to optimally place the nodal positions and minimise the issues associated with reticulated grids in general and problems specific to the V-Stripes detailed above.

Combined assembly strategies

The presented V-shaped stripe generations lead to a powerful tool when the different systems are combined, or used with additional constraints. Reticular structures that consist of combined regular and irregular grids, in order to achieve a minimisation of different folding angles for the entire structure, can be imagined. Meshes can be utilized that were not only generated through form-finding techniques but also took into consideration a reduction in the production time, by reducing the number of folds needed.

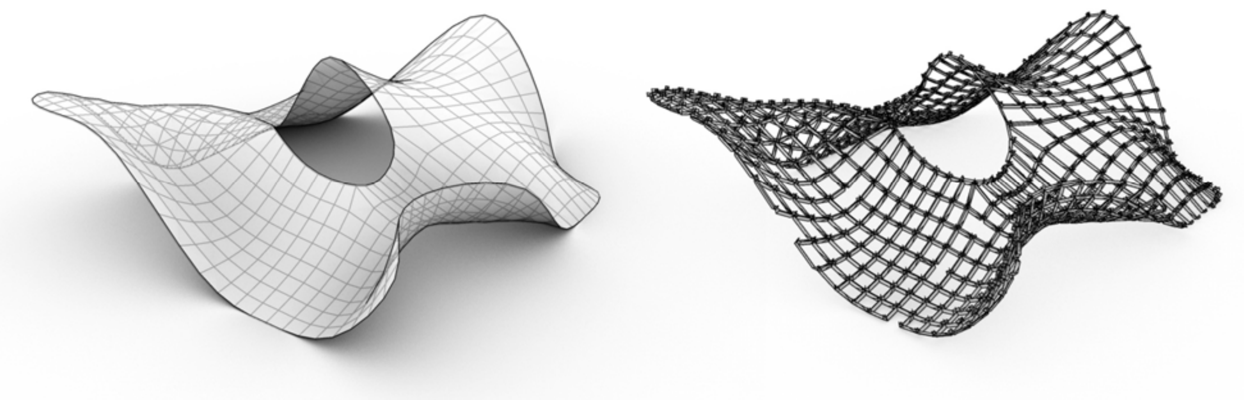


Figure 16: a surface approximation from a mesh with irregular mesh based stripes

Besides the combination of different systems approximating one given Surface, V-shaped stripes can also be used to approximate double layered surfaces with closed stripe systems (Maleczek; Geneveaux; 2011). This strategy could be used for the creation of doubly curved sandwich panels at a small scale, and for the creation of double layered building structures for freeform spaces.
An ongoing area of research is the investigation into the structural abilities of the different systems. This exploration, in combination with the production issue, will extend the possibilities of the presented structures, and seem to be very promising for the future.

Another recently started research on this system is the use of curved folded stripes, to minimize the number of folds, and extend the structural abilities of this system (Fig. 17).
Recent findings such as this show that the vast range of potential of developing and using such systems is just being uncovered and will only continue with further research.

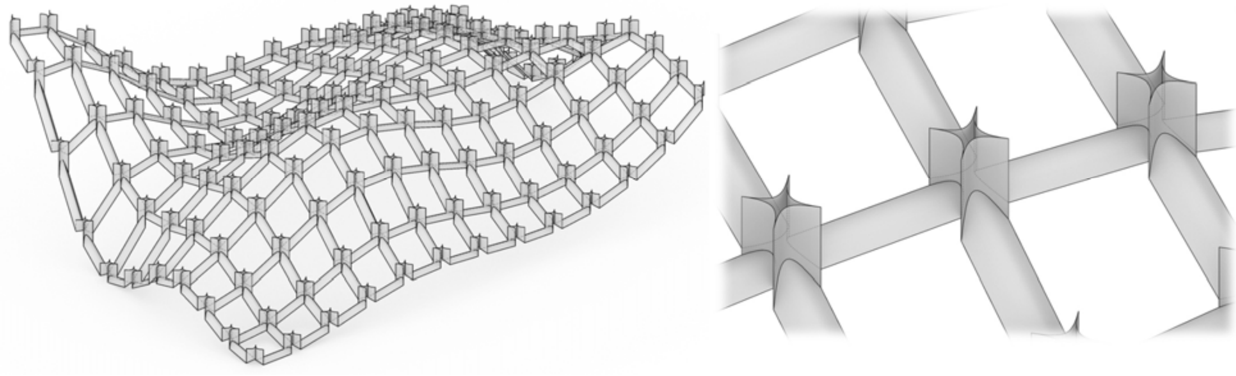


Figure 17: a surface approximation with curved folded stripes

List of references

Tomohiro T., 2009: One-DOF Cylindrical Deployable Structures with Rigid Quadrilateral Panels. In: Proceedings of the IASS Symposium 2009, 2009, pp. 2295-2305.

Klett Y.; Drechsler K., 2011: Designing Technical Tessellations. In: Wang-Iverson P et al. (eds), 2012: *Origami 5*, London, CRC Press, pp 305-322.

H. U. Buri,2010: "Origami–folded plate structures", PhD Thesis, École Polytechnique Fédérale de Lausanne, Laboratoire de Construction en Bois, 2010

Maleczek R. 2010:*"Linear folded parallel stripe(s)"*, Computational Design Modeling,Proceedings of the Design Modelling Symposium Berlin 2011; Gengnagel et al. ; Springer Berlin 2011, ISBN 978-3-642-23434-7.

Maleczek, R.; Geneveaux, C.; Ladinig, H.,2012: Linear folded meshbased stripes, for IASS-APCS Conference2012, Korea;Seung Deog Kim; p.139; ISBN 978-89-968907-1-3;

Maleczek, R.; Geneveaux, C.2011: *"Open and closed linear Folded Stripes"*, Taller,Longer,Lighter;IASS-IABSE Symposium, London2010.

Delarue J-M, 1987: "*Constructions plissées – rapport final de recherche*", Ecole d'architecture Paris-Villemin, Paris, 1987, pp. 1-78.

Demaine E D., O'Rourke J.,2007: "*Geometric Folding Algorithms, Linkages, origami, Polyhedra*", Cambridge University Press,pp.29-147.

Shepherd P, 2011: "The Benefits of Subdivision Surfaces for Complex Geometry",for IASS Structural Morphology Group International seminar September 2011; Conference Proceedings.

Williams, C. J. K., 2001. "*The analytic and numerical definition of the geometry of the British Museum Great Court Roof."* In: Burry, M., Datta, S., Dawson, A. and Rollo, A. J., eds. Mathematics & design 2001. Geelong, Victoria, Australia: Deakin University, pp. 434-440.

Progress towards Multi-Criteria Design Optimisation using DesignScript with SMART Form, Robot Structural analysis and Ecotect building performance analysis

Robert Aish, Autodesk, Al Fisher, Buro Happold,
Sam Joyce, University of Bath, Andrew Marsh, Autodesk

Abstract

Important progress towards the development of a system that enables multi-criteria design optimisation has recently been demonstrated during a research collaboration between Autodesk‟s DesignScript development team, the University of Bath and the engineering consultancy Buro Happold. This involved integrating aspects of the Robot Structural Analysis application, aspects of the Ecotect building performance application and a specialist form finding solver called SMART Form (developed by Buro Happold) with DesignScript to create a single computation environment. This environment is intended for the generation and evaluation of building designs against both structural and building performance criteria, with the aim of expediently supporting computational optimisation and decision making processes that integrate across multiple design and engineering disciplines.

A framework was developed to enable the integration of modeling environments with analysis and process control, based on the authors‟ case studies and experience of applied performance driven design in practice. This more generalised approach (implemented in DesignScript) enables different designers and engineers to selectively configure geometry definition, form finding, analysis and simulation tools in an open-ended system without enforcing any predefined workflows or anticipating specific design strategies and allows for a full range of optimisation and decision making processes to be explored.

This system has been demonstrated to practitioners during the Design Modeling Symposium, Berlin in 2011 and feedback from this has suggested further development.

1 Introduction

The optimum design of buildings is a recurring challenge to architecture and engineering teams. But to begin with we need to define what we mean by design optimisation? „Design Optimisation‟ is a really a shorthand for „performance satisficing‟, that is the design of buildings to effectively satisfy multiple potentially conflicting performance criteria.

Historically, there have been three approaches (Figure 1):

- Post-rationalisation: a Building concept form is proposed by an architect and then „after the fact‟ the design is analysed, its performance is evaluated and the building geometry and engineering implementation is rationalized, with the objective of improving the performance, while minimizing the change to the original building form or design concept. [For example: Foster + Partners‟ London City Hall building.. where a „pebble‟ shaped building concept was rationalized into a series of sheared cone constructions]

  Effectively: Design -> ***Solution***

- Pre-rationalisation: Before the form of the building is defined, there is agreement amongst the design team to use particular architectural geometry or construction techniques that are thought to provide an optimum solution. The building form is proposed by the architect within these constraints [For example Foster + Partners‟ Sage Performing Arts Centre, Gateshead, where the use of torus patch geometry was predefined in order to optimize the facade fabrication process] (Whitehead and Peters 2008)

  Effectively: Solution -> ***Design***

- Embedded rationality: The engineering performance assessment and the form generation algorithm are combined into a single design optimisation process [For example Foster + Partners‟ Roof for the Great Court of the British Museum, where the optimum form of the roof geometry was arrived at by computation] (Williams 2001)

  Effectively: Solution <-> ***Design***

Figure 1. Examples of alternative forms of 'design rationalisation'

We can see that existing approaches of pre and post rationalization have produced some interesting results but are essentially expedient. This is due to the fact that in both cases some predefined conditions have been applied that in most cases lead to constraints in deriving the optimal form. As such, it is generally accepted that the most appropriate approach to truly open ended design optimisation is through embedded rationality.

This acceptance comes from the understanding that buildings are collections of closely coupled subsystems, such as the envelop, internal spatial topology, structure, building services, occupancies and energy transfer systems, each with their own engineering discipline and performance criteria. To create an optimal building, there are important interactions to be considered and trade-off‟s to be made within and between these subsystems and the derived or emergent whole. Each subsystem may be evaluated in terms of its capital and running costs. Therefore single criteria optimisation is inappropriate.

There are also practical issues for designers to gain access to design optimisation tools. A „design-centric‟ approach is based on augmenting generative design tools with easy to use analysis and optimisation add-on‟s, but the downside is that these add-on‟s often reflect the assumptions of the add-on creator and may be restricted by these assumptions, while at the same time such generality may not be matched to the specific design problem being tackled.

Conversely specialised software tools, created by advanced scripting, can be used to connect programs and control complex optimisation with decision making processes: the downside of such specialist (or project specific) tools are that they are often: (a) only applicable for use on well-defined problems in complex large scale projects (b) are not sufficiently general or reusable (c) require considerable insight on the part of the users and (d) are therefore not practical for the use by non-experts.

We can chart the evolution from conventional computer aided design to design optimisation, as follows:

1. ***CAD*** ... early CAD tools were developed to offer a digital implementation of conventional analogue design media, such as.. sketching, drafting, modeling, which required the designer to „manually‟ construct the design configuration.
2. ***Generative design tools***... (Figure 2) changed the design paradigm from analogue conventions and the direct construction of the design by the user. Instead, the designer indirectly controls the generative process by:
   - developing the set of constructive/generative rules
   - defining the value of the set of „design driver‟ variables
   - interpreting the resulting generated design

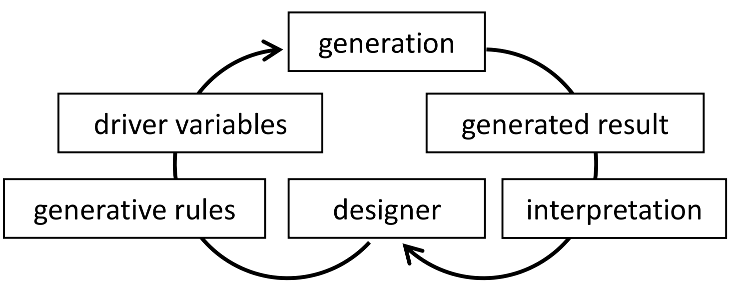


Figure 2. Generative design

3. ***Engineering analysis tools***.. (Figure 3) here the designer controls the process by:
    - selecting the analysis tools
    - defining how the design configuration is idealised into a form suitable for the chosen analysis methods
    - interpreting the resulting performance analysis

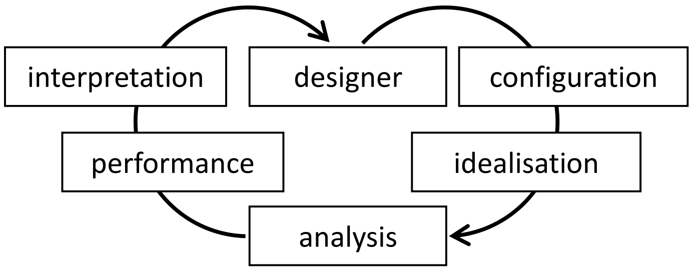


Figure 3. Engineering analysis

4. ***Design optimisation***.. (Figure 4) Combines generative mechanisms and analytical /evaluative mechanisms into a single iterative process, in which the performance analysis is a direct input into the generative process. The designer controls this process by:
    - defining a single „utility" measurement to compare different designs [usually based on some weighted combination of different performance variables]
    - specifying the mechanism to automatically generate new candidate configurations [either by generating new combinations of driver variables or by modifying the generative rules]

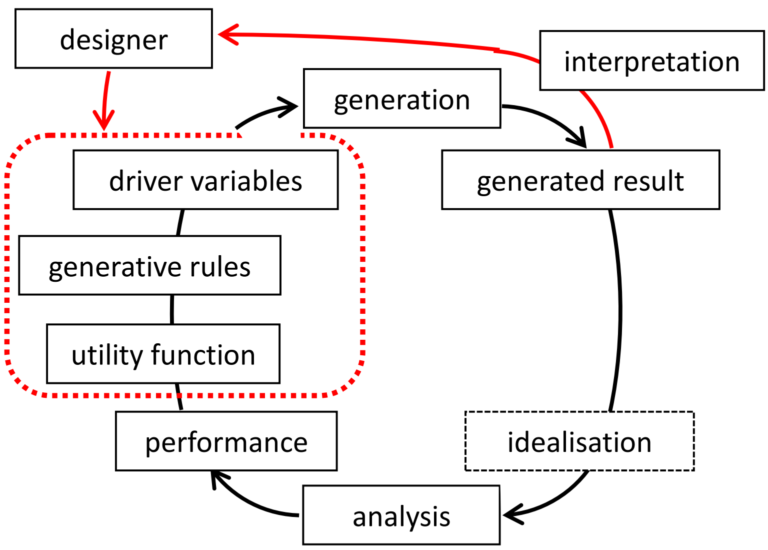


Figure 4. Design optimisation

In the progression from CAD to Design Optimisation, we see increasing levels of indirection as the designer progressively removes himself not just from the direct act of designing, but also from the evaluative loop. He moves from „doing" to the far more strategic role of „controlling".

So in summary, design optimisation depends on some or all of the following:

- A generative process (to construct the design alternatives), which may be explicitly driven by identifiable design variables
- A number of evaluative processes (to evaluate the performance of the different subsystems)
- A fitness function to combine all performance criteria into a single fitness measure
- A manager process that:
    - initiates the generative process with some initial values for the design variables
    - drives the evaluative processes
    - executes the fitness function
    - decide whether an optimum design has been produced and if not
    - refines the values of the design variables
    - and continues the iterative optimisation process

Any one of these processes may use a human designer or engineer, or a computer based application. The manager process may include numeric optimisation techniques, genetic algorithms or neural networks for decision support. Also this process is in many instances a hierarchy of systems and subsystems, each with their own internal decision making and change propagation logic.

2 Current Research:

Progress towards the development of a multi-criteria design optimisation system has recently been demonstrated during a joint research collaboration between the Autodesk"s DesignScript development team and the engineering consultancy Buro Happold. This research builds on the authors" previous work, including the development of domain specific end-user programming languages, (Aish, 2011), the use of genetic algorithms for structural optimisation (Evins, Joyce et al, 2012) (Shrubshall and Fisher, 2011) and the use of a physics solver to optimize geometric configuration of facade planar quads (Attar, Aish, Stam et al 2009).

This project aimed to build on this research by integrating the following technologies chosen for their broad but practically applicability informed by the authors experience in the industry.

- Generative Building Design
    - using associative parametric modeling in DesignScript
    - algorithmic form finding using Buro Happold‟s „SMART Form‟‟ software integrated into DesignScript
- Engineering performance analysis
    - structural analysis using aspects of the Robot Structural application integrated with DesignScript
    - environmental analysis using aspects of Ecotect integrated into DesignScript
- Optimisation management process: where change logic and control is automatically propagated by DesignScript to re-compute:
    - Underlying architectural geometry
    - SMART Form form finding
    - Robot structural analysis and member sizing
    - Shading device geometry creation
    - Ecotect insolation analysis of the shading device geometry

It is important to note that this particular sequence (geometry, form finding, structural analysis, shading geometry and insolation analysis) was a particular modeling sequence that was appropriate to the demonstration project. Different projects could have different modeling sequences and are equally well supported by this system.

The demonstration project was the design of a roof to cover the ruined shell of a gallery at the rear of the University of the Arts in Berlin (Figure 5).

Figure 5. The site for the demonstration project: the gallery at the rear of the University of the Arts in Berlin

The goal was to conceive of a system where the objects produced by generative building techniques or any other means (stochastic‟), were direcly linked to their corresponding analysis representations and results objects.

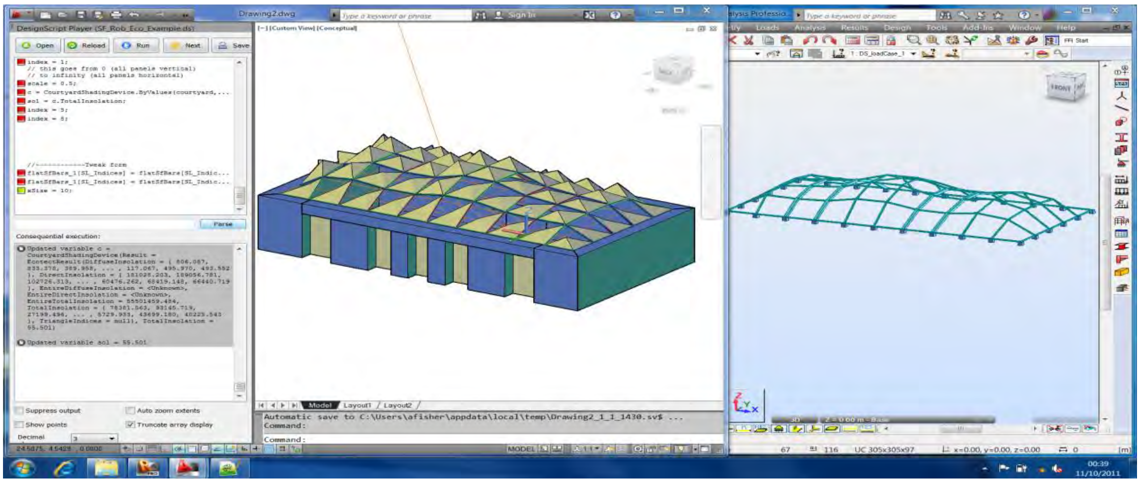


Fig. 6: DesignScript environment showing:
Top Left: DesignScript source code window
Lower Left: DesignScript 'consequential' execution window
Center: DesignScript 'model' with SMART Form, Robot and Ecotect model
Right: Robot structural analysis application driven remotely via DesignScript

3 Implementation

The implementation depended on the integration of Robot, SMART Form and Ecotect into DesignScript (Figure 3). This was achieved by developing special DesignScript classes for each of these engineering applications. The methods in these DesignScript class made calls into external methods and functions in the respective host applications using the DesignScript Foreign Function Interface (FFI).

The following table (Figure 7) describes the implementation of the different plug-in‟s using the DesignScript Foreign Function Interface (FFI).

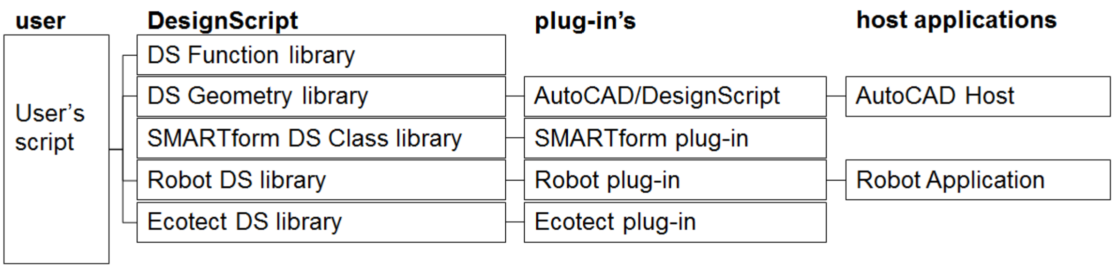


Figure 7. The DesignScript application architecture with the ability of a single script to execute different plug-in's on multiple host applications.

The DesignScript Foreign Function Interface (FFI) is exactly the same technology that DesignScript uses to interface to the CAD host application (currently AutoCAD).

3.1 Robot integration

Implementation

The integration with the Robot structural analysis application was implemented as a series of "Structural" classes directly accessible and instantiated by users. The connectivity of the instances of these "Structural" classes builds graph-network relationships, with helper functions to enable „dumb" geometry to be promoted to structural elements.

The elements of the structure to be calculated are then passed into an "Analysis" object. The intention is that this "Analysis" object allows the user more direct control over the execution of what could possibly be a computationally heavy task. This structural "Analysis" object then creates a collection of structural "Result" objects corresponding to the collection of input structural objects. In this way the structural analysis could be used in both associative programming and (in future) in imperative programming. These "Result" objects can be interrogated for their analytical information both at a model level (for example, the overall deflection) and at an element level (for example, shear stress at a point along a beam).

User centric orientation

An important aim for the Robot integration was to enables a non-engineer to develop a structural model and to make a reasonable interpretation of its performance. Support for the non-specialist user included providing intuitive methods to help the user give reasonable values for complex structural settings (for example, bar gamma angles defined by the direction of the surface normal and parametric section definitions).

Another way that the Robot integration supported the non-specialist user was to provide more holistic measures of performance, such as "material utility" (maximum analysed stress/allowable stress) which can simply show if an element is unsafe (over 1) and if not how well the element is used (0-1). This type of holistic measure is complimentary to the more conventional indicators of structural performance such as Bending Moments and Von Mise Stress. The "material utility" results can be interrogated both in the generated Robot model as well as visually displayed within the Design Script environment (Figure 8).

3.2 SMART Form integration

Implementation

Much like the Robot implementation described above, the integration of SMART Form enables SMART Form classes to be instantiated directly within the DesignScript environment. The form finding process works by defining individual geometrical entities as „bars" with elastic properties. So again a graph-network relationship of nodes and bars is created and stored. This symmetry with the Robot analysis means that once a structural graph has been defined by a user it can be mapped directly from SMART Form to Robot or *vice versa*. An iterative process of non-linear structural analysis is then performed to find the equilibrium geometry for the given structural properties (elasticity, member slack length/pre-stress) and boundary conditions.

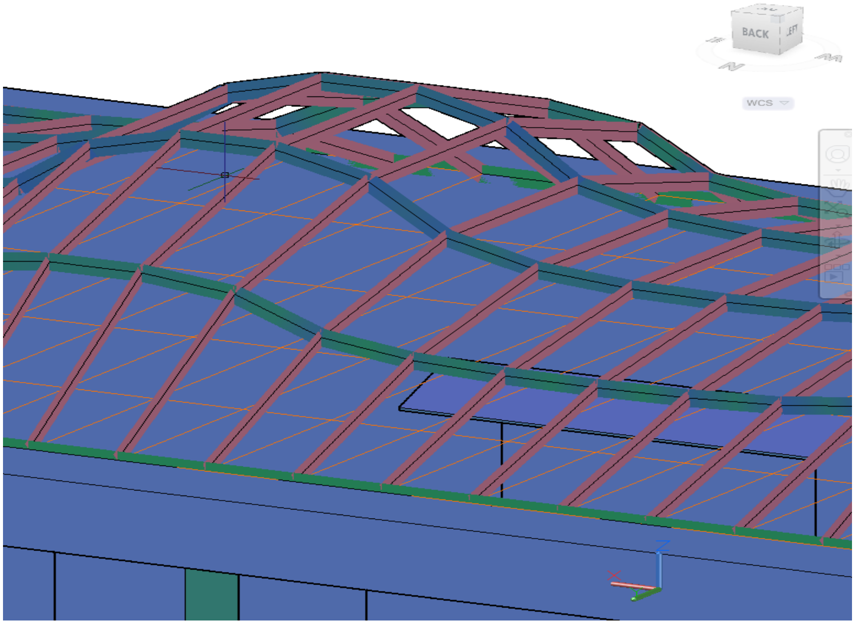


Figure 8. The output from Robot Structural analysis displayed in DesignScript, showing the ultilisation of the structural members, colour coded green to blue for under utilised and red for over utilized.

Design intent

Defining and exposing the structural properties inherent within the DesignScript environment allows users to manipulate the values and thus sculpt a desired form for their design. Here the structural performance criteria of minimum energy for the system is persistent within the model and is used to drive the equilibrium form. Thus these parameters can be manipulated, through moving boundary supports, changing pre-stress, introducing heterogeneity in the stiffness distribution *etc.*, to satisfy additional design and analysis criteria, as the demonstration workflow illustrates below (figure 9).

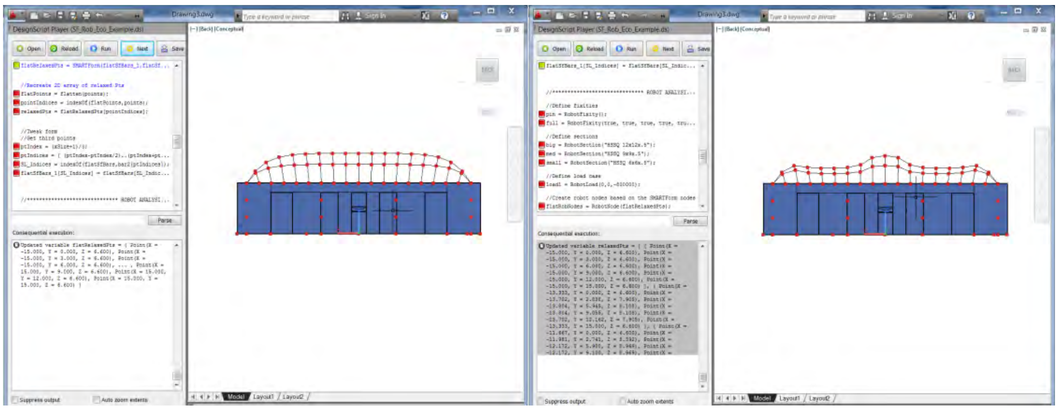


Figure 9. SMART Form executed within DesignScript. Left: Relaxation of a network of bars. Right: Sculptural manipulation of the form enabled through varying stiffness properties

3.3 Ecotect integration

The Ecotect plug-in in this instance focused on solar design and analysis. The calculation of instantaneous incident solar radiation is relatively straightforward as it involves just a single sun position and everything can be readily solved geometrically. However, of significantly more use to a designer are cumulative results such as the total collection over the whole year or just for summer. This significantly increases the calculations required, making these potentially very computationally expensive as they are highly dependent on the geometric complexity of both the model and any potential obstructions that surround it. Thus, rather than provide simple, high-level functions that return results for a given set of date, time, location and geometry inputs, the aim in this work was to provide scope for experimentation and usage patterns not envisaged by the plug-in developers, as well as support interactive design feedback, which ideally requires calculation results as close to real-time as possible.

Achieving fast results on-demand required a tight integration of the calculation process within the DesignScript environment, and a multi-step approach that allows for the optimized caching of reusable results. This also meant exposing the individual services within Ecotect that dealt with geo-location of the site, accurate determination of solar position, detailed shading/overshadowing calculations, access to hourly weather data files, and then the incident solar radiation functions that coordinate all this information into useable values. The various DesignScript classes required are shown below (Figure 10).

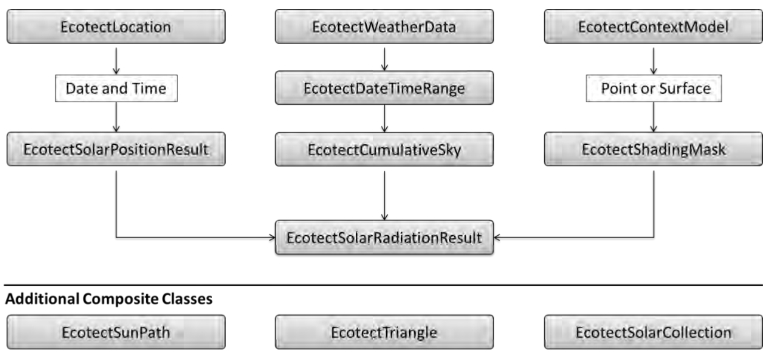


Figure 10. An outline of the DesignScript classes exposed from Ecotect.

Once a weather data file and location have been selected, a significant amount of solar information can be pre-calculated and cached for subsequent (re)use. Similarly, whilst local overshadowing on a potentially animated building model will vary significantly, the use of a static context model of site obstructions means that optimizations such as spatial trees can be pre-calculated to significantly speed up the calculation of global overshadowing.

The most significant improvement in computational performance is the move from multiple individual solar position calculations to the use of a sub-divided sky model. This is a well-known technique (CIE 1994) but the innovation here is the consistent separation of each component of the calculation, several of which can be pre-calculated and/or augmented from global model information, either just-in-time or while the system is idle (Figure 11).

In it, the diffuse and direct solar energy distributions need only be calculated once when the weather file and location are initially set. Similarly, the cosine law distribution for a flat surface can be very quickly determined from a pre-calculated spherical distribution, using the surface orientation to index it appropriately. Also, if the same model is to be used in a series of interactive analysis, this approach allows a script to save calculated obstruction and reflection masks for individual surfaces, groups of objects or even the whole model to disk for re-use in each subsequent analysis.

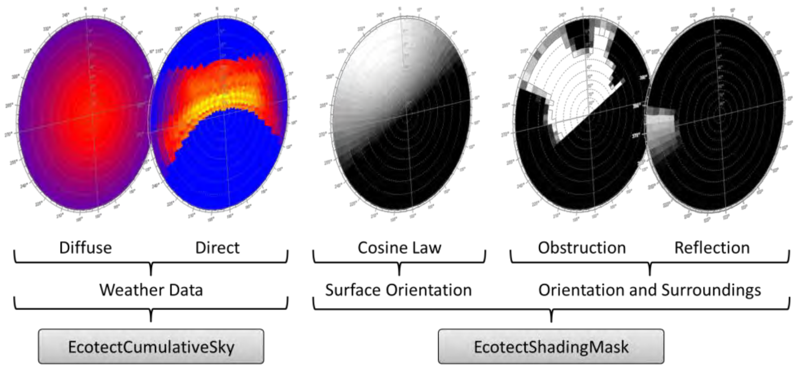


Figure 11. The use of a sky subdivision model and the separation of each component of the solar radiation calculation.

This integration of SMART Form, Robot and Ecotect is part of a strategy to integrate a number of design tools (geometric, generative and evaluative) into DesignScript. Essentially DesignScript is extensible and using the „Foreign Function Interface" (FFI), classes in external DLL"s can be exposed as DesignScript Classes.

4 The Current Application

A demonstration process was developed based on the capabilities of the current system and this formed the core of the workshop on DesignScript at the Design Modelling Symposium, Berlin, 2011. The intent was to present a multi-scalar analysis and optimisation process which utilised functionality from all of the plug-ins developed and with the DesignScript language as the unifying technology.

The overview of the process was as follows:

1. Establish Site constraints (model existing building shell)

2. Make regular rectilinear grid over the plan courtyard with a parametric number of elements in each direction

3. Generate a relaxation model derived from the geometric model, with fixed boundary nodes on the edge of the grid and liner spring elements in place of the lines.

4. Relax the model with a negative gravity force to produce an efficient structural form

5. Identify the relaxed grid cells and generate shading panels associated to the grid unit

6. Convert the initial geometry into solar analysis panels.

7. Extract sun path information and analyse the solar panels

8. Orient the panels based on this information

9. Obtain the overall insolation incident on the courtyard of the space

10. Modify the gravitational force of the relaxed grid (step 4) to influence the insolation on the floor by reviewing the updated insolation analysis values

11. Develop a steel structural model with uniform sections based on the relaxed grid with the same boundary conditions as the relaxation model.

12. Check the maximum stresses in each of the beams

13. Size up any failing beams and down any under-stressed beams in proportion to the amount they are off the ideal utilisation of the material

14. Resize the base grid (step 2) and after the auto update of all the other modeling and analysis systems, review whether the steel weights significantly change

This initial research demonstrated the capability of the system to support the design decision process informed by appropriate and reliable performance criteria. The ability to nest and reorder generative and analytical processes within the same overall computation design environment is another important feature of the system. This allows the generation of configurable hierarchies comprised of interrelated geometry generation, analysis and decision making processes.

The system was initially tuned by the user and then subsequently by basic implementation of a simple Newtonian goal seeking algorithm. The quality of actual optimisation processes can be refined with more time and is the subject of the next research phase.

When demonstrated at the Berlin workshop, DesignScript with its set of plug-in‟s was generally regarded with interest as a system with the capability for practical performance driven design. The workshop participants spanned a broad range of architectural and engineering experience and a number of the participants were able to take this model as a starting point for their own exploration, including re-orienting the hierarchy of the design logic towards their own intentions.

5 Future Research:

Based on the feedback from the DesignScript workshop…

There are number of interesting opportunities on the horizon for the next round of research:

- ***Imperative Programming***: With imperative programming being added to DesignScript, the ability for practitioners to develop their own decision making and optimisation routines exists. We are preparing for a second stage in the joint research collaboration between the Autodesk DesignScript development team and Buro Happold. Imperative programming will enable a general purpose genetic algorithm to be developed for DesignScript.
- ***Options Language and Cloud computing***: The DesignScript is being extended with a special „Options‟ language, which can be to control the generation of multiple alternative design solution using cloud based parallelism. Design optimisation, and specifically genetic algorithms require large number of solutions to be generated. So this approach will be important in future design optimisation.

Conclusions:

The Multi-Criteria Design and Optimisation of buildings poses both an interesting technical challenge and potentially a powerful means by which to drive design towards better performance.

This paper shows a conceptual approach to supporting optimisation within a single computational design system which crosses traditional discipline boundaries and can therefore address issues of optimization which are inherently multi-disciplinary.

While the idea of design optimisation has been discussed in the research literature, it has not been widely used in practice, mainly because tools based on optimization have not been widely available or indeed available in forms which are easily accessible to practitioners.

Therefore, one important future challenges for software developers is to make optimisations tools more accessible and more easily used by practitioners, but without compromising the rigor of use that is required to achieve valid results. The anticipated increase in adoption of optimisations tools has the potential to bring substantial benefits not just to architects and building engineers but to the users and owners of buildings and thereby address wider economic and sustainability concerns.

References

Whitehead, H. and Peters, B. (2008). Form and Complexity. In Space Craft: developments in Architectural Computing. (ed) D. Littlefied, RIBA Enterprises.

Williams, C. (2001). The analytic and numerical definition of the geometry of the British Museum Great Court Roof. In Mathematics and Design (eds) M. Burry, S. Datta, A. Dawson, and A. J. Rollo, 434-440, Deakin University, Geelong, Victoria 3217, Australia, 2001.

Aish, R. (2011). DesignScript: origins, explanation, illustration, Design Modeling Symposium, University of the Arts, Berlin

Evins, R., Joyce, S. et al (2012). Multi-objective optimisation: getting more for less by design, Proceedings of the Institution of Civil Engineers, Civil Engineering Special Issue 165.

Shrubshall, C., Fisher, A. (2011). The Practical Application of Structural Optimisation in the Design of the Louvre Abu Dhabi. In: Proceedings of the International Association for Shell and Spatial Structures Symposium. London.

Attar, R., Aish, R., Stam, J., et al (2009) Physics-based generative design, CAAD Futures.

Commission Internationale de l'Eclairage Guide to recommended practice of daylight measurement (1994). (ed) J D Kendrick, Vienna: Commission Internationale de l'Eclairage.

Multi-objective optimisation: higher ‘performance’ for lower ‘cost’

Ralph Evins, MEng. Buro Happold and University of Bristol.
Sam Joyce, MEng. Buro Happold and University of Bath.
Philip Pointer, Buro Happold.; Shrikant Sharma, Buro Happold.
Ravi Vaidyanatha, Imperial Collage.; Chris Williams, University of Bath.

Abstract

Multi-objective optimisation can help civil engineers achieve higher performance for lower costs in their designs. This is true whether ‘performance’ applies to structural strength or energy use, or whether ‘cost’ measures financial outlay or occupant satisfaction: if it can be quantified it can be optimised in some form. By exploring trade-offs between conflicting objectives and constraints, multi-objective optimisation enables informed decision-making. This paper outlines the principles and benefits of multi-objective optimisation and the means of implementation. The complementary aspects of parametric modelling and optimisation are discussed as an aid to the flexible design of buildings and structures. A range of real design problems are considered, including structural and environmental examples.

Keywords

Optimisation, Multi-objective optimisation, integrated design, parametric design.

Introduction

Current civil engineering design practice is epitomised by an ‘informed trial-and-error’ approach to optimisation: ‘designs are still optimised mostly through a manual iterative process’ (Roy et al., 2008). There is often the potential to add significant value by using more explicit methods to explore the design space. Other industries (e.g. aerospace) have long taken advantage of a more rigorous approach to engineering design optimisation and this trend is now beginning to take hold in civil engineering. Academic examples cover a wide range of applications, including optimising structural design (Koumousis and Georgiou, 1994), geotechnical performance (Zolfaghari et al., 2005), building form (Marks, 1997), fabric properties (Wang et al., 2005), heating, ventilation and air- conditioning systems design (Fong et al., 2006) and control (Huang and Lam, 1997).

All practicing civil engineers will recognise the description of ‘a complex, multi-disciplinary engineering activity that requires making difficult compromises to achieve a balance between competing objectives’ (Ren et al., 2011). At a fundamental level, there is a need to consider all sub-domains of the field and their impact on the overall design. At a broader level, there is a need for a holistic consideration of design and context. For example, the design of a new office building might address the impact of business practice on space requirements, commuting distances in relation to site selection and mixed-use development to allow a site-wide energy scheme to improve energy efficiency.

Applying multi-objective optimisation methods requires careful consideration of the system in question. It is not practical to consider all sub-systems and variables simultaneously; the formulation of system boundaries such that some things are varied while others are held constant is of critical importance. The system to be optimised is defined by objectives, variables to adjust and constraints that must be maintained.

Multi-objective optimisation

Consider a generic structural problem concerned with strength and cost. These two objectives are in conflict: a solution may be ‘cheap but weak’, ‘strong but expensive’, or anywhere in between. These two objectives are shown on the axes of Figure 1 (by convention, both are to be minimised). There exists a set of possible solutions all of which are optimal for some trade-off between strength and cost. The purpose of multi-objective optimisation is to find this set of optimal solutions (the yellow points in Figure 1), referred to as the trade-off front or Pareto front.

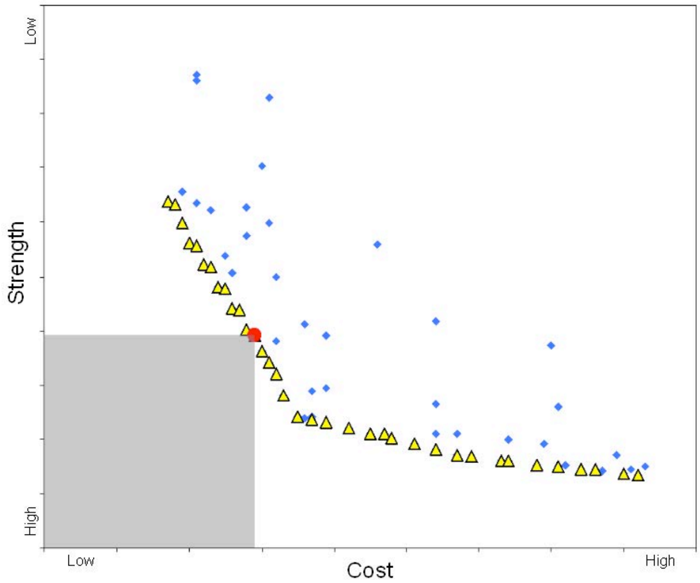


Figure 1: An example 'trade-off front' or 'set of optimal points'.
Yellow solutions are optimal; blue are not.

What makes optimal solutions distinct from non-optimal alternatives? For optimal solutions, there exists no other solution that is better in all objectives; that is, in this example, there is no solution that is both cheaper and stronger. If such a solution existed, it would clearly be preferred. In Figure 1, this is illustrated for the red solution: there are no points in the grey area, so this solution is part of the optimal set.

It is often not possible to say in advance where on the trade-off front it is most desirable to be. It is not always possible to specify the importance of each objective or to combine them into a single objective by applying weights. Exploration of the shape of the trade-off front allows informed decision-making regarding marginal benefits. The example in Figure 1 contains a distinct kink – the marginal increase in strength for a unit increase in cost changes dramatically at this point. The aim of multi-objective optimisation is to discover the entire trade-off front; solutions should be well distributed along the front rather than occupying only a small niche.

There are many means of accomplishing the goals of multi-objective optimisation. The most well-known is the genetic algorithm, a form of evolutionary computation; other methods operate along similar lines, for example differential evolution, evolutionary strategies and genetic programming. These algorithms were inspired by Darwinian evolution or 'survival of the fittest'. They mimic competition for survival among a 'population' of many 'individuals', each corresponding to a particular solution to the problem. Each individual possesses a certain 'fitness', which is measured against the objectives. Competition is enforced by eliminating individuals of predominantly poorer fitness, thus causing the fitness of the population to improve over time.

For a problem with a single objective, fitness can simply be proportional to the performance of the solution against the objective. Fitness assignment is more complicated for the multi-objective case. A variety of methods exist, generally using the principle of distance from the current trade-off front. One popular technique assigns the highest rank to solutions in the overall trade-off front, then removes these from contention and recalculates the next front, which is assigned the next rank (Deb et al., 2002).

As well as a means of preferring solutions of higher fitness, individuals must also be encouraged to change over time in order to fully explore the problem domain. The genetic algorithm achieves this using two operators that mimic biological processes: the crossover between individuals and random mutation. The former involves splicing characteristics of two individual into new combinations to allow inheritance of good characteristics; the latter randomly alters values of an individual, in order to explore the search space more widely. By repeatedly performing the process of alteration and selection, the population improves with each subsequent 'generation'; an 'elite population' can also be derived, consisting of the best individuals from all generations. A schematic illustration is shown in Figure 2.

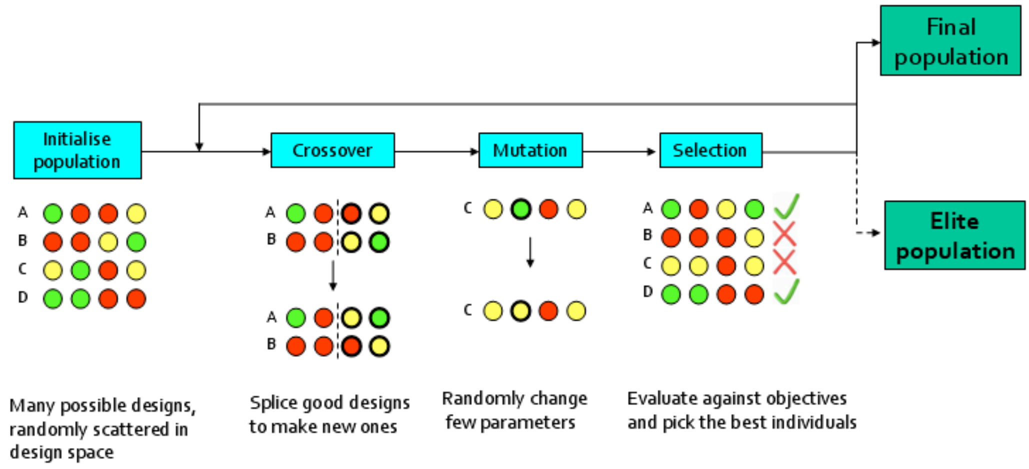


Figure 2: Schematic illustration of a simple genetic algorithm.
The operators are illustrated for four individuals a–D

There is great flexibility with regard to how to encode a problem for solving by such algorithms. Variables can consist of binary, integer or real numbers, or tree-structures that can represent operations, ordered graphs or computer programs. These algorithms have been implemented in many programming languages and platforms and both free and commercial packages are available. The algorithms can be configured to repeatedly call external programs that perform simulations for particular sets of variables and return output values that quantify performance against objectives.

Parametric design

Parametric design is a developing term used to encapsulate a method of design that involves using computational processes to define form. Its role within the design world is growing as firms are becoming increasingly aware of the benefits that automated techniques provide over other approaches. The capabilities of the computer provide a significant step change in the efficiency of the design process. It is now recognised that parametric design could further supplement current techniques by providing more holistic and adaptable tools that are aligned both with computational process and natural design thinking.

Computer-aided design (CAD) as a design tool has many advantages over traditional hand-draughting methods as it provides conveniences such as undo functions and cutting and pasting of information. Whereas these increase the speed of the drawing process, the CAD file is essentially a digital reproduction of conventional draughting information. The main issue with this is that the information is that of complete exception, where every mark is unique and the model has no intelligence about the relationships between items (Coenders, 2009).

Parametric design has the potential for a greater impact on the design process by capturing the design rationale rather than a static design drawing. It uses associative relationships to ensure that the logic of the design is embedded within the model – any readjustment and thus regeneration of the design uses this to automatically update the final output. The key concept is to create geometry that has logical associative links, such as the position of a beam being dependent upon the top points of the columns by which it is supported. Normally, this is implemented by way of a hierarchical system where basic geometry is built up and developed until a complex representative model is produced. This is done using logical and geometric operations following computer programming principles and CAD software capabilities respectively.

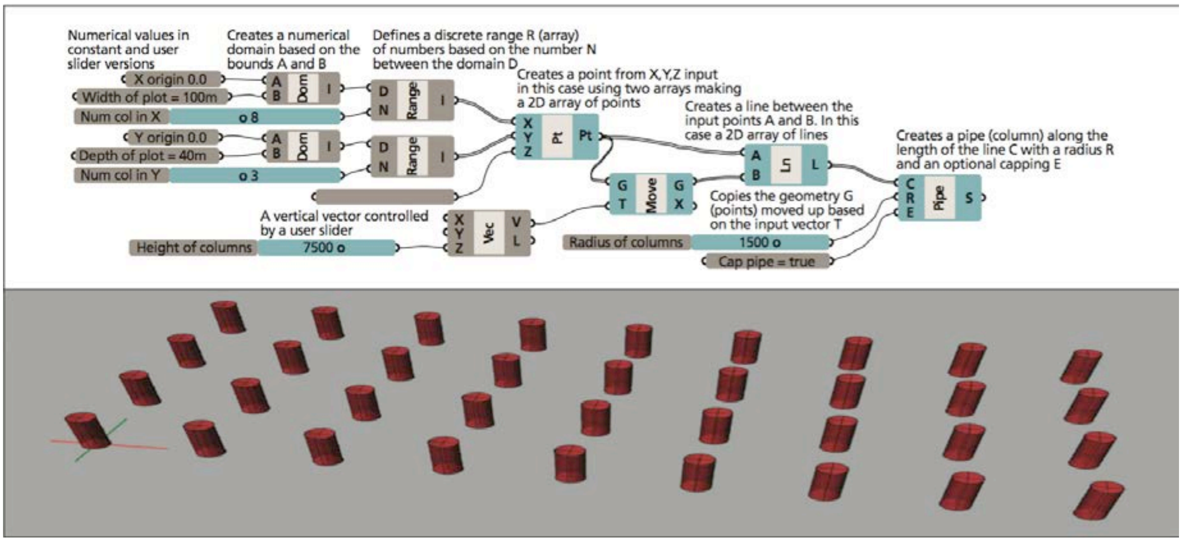


Figure 3: Example of an associative model visualised in parametric software. The model produces a regular grid of columns with number of columns in X and Y as well as their height and width as adjustable parameters. The output of the associative model for one configuration of parameters is shown below the model sketch.

The resulting power of this system over conventional CAD is twofold. First, the generation of a model can be linked to input values or parameters (hence the name). For example, a series of input parameters can be identified for various properties of a design such as the number of floors in a building, the length of shading overhang, the ratio between member length and diameter and so on. These parameters can then be modified, allowing for a high level of design flexibility, with options generated and modelled sequentially. This allows the user to design flexibility into a model where values are uncertain or variable.

The second feature is that the inbuilt logic of the design system does not change, irrespective of parameter values. The model will adapt based on the established rules and new variants produced with a change in parameters. The result is a ubiquitous and adaptive design system that can lever computational power to offer more possibilities in less time in comparison with conventional CAD.

Whereas the parametric design approach has been in existence for some time, the process has not yet been refined in its entirety. For example, parametric modelling was adopted for the generation of the roof of the British Museum in 2001, but this required highly skilled specialists with programming skills to make designs in this way (Williams, 2001). The complexity of this approach presents considerable barriers for employment of parametric methods to all but highly specialist teams.

The parametric approach has been made more accessible by computer programs that enable the creation of associative models in more intuitive ways. Generative Components (www.bentley.com/getgc) and Grasshopper (www.grasshopper3d.com), produced by Bentley and McNeel respectively, are implementations of parametric CAD software. They use network graphs to aid in the creation and visualisation of the associative links that govern the design (Figure 3) as well as allowing real-time update of the design model as changes are made to parameters. This increased accessibility means that it is now practical for most companies to implement this method within their design processes, for either discrete elements or an overall design. This has translated into a greater adoption of parametric design processes on real projects such as the 2010 Aviva Stadium in Dublin, (Shepherd and Hudson, 2007) and others (Hesselgren et al., 2007).

It is worth noting that, at the time of writing, the two principal programs for the generation of parametric designs are essentially free. It is thus now considerably more practical and expedient to learn and implement parametric modelling within design practice.

The model produces a regular grid of columns with number of columns in X and Y as well as their height and width as adjustable parameters. The output of the associative model for one configuration of parameters is shown below the model sketch

Application: structural design

The first case study shows the use of parametric design to explore initial structural solutions for a large roof canopy. The positions of the truss elements were defined in plan based upon the requirement for coordination between the glazed facade and the roof. The structural elements needed to be situated in a predefined volume between roof and ceiling cladding. The aim was to produce an efficient truss, such as the one shown in Figure 4. Here, the secondary trusses can be seen as those that span along the short length of the roof and the primary trusses are those that intersect the secondary tresses (typically twice) and follow the glazing line of the building. For this specific design, the section sizes were already determined by previous constraints.

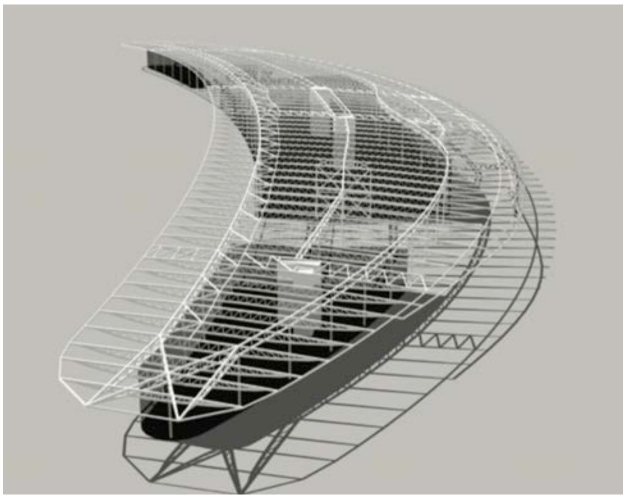


Figure 4: Perspective of truss geometry within the ceiling cavity.

The design was driven by two principal parameters, both controlling the spacing of truss bays, which are defined here as one 'X' arrangement of webs between the chords. The first parameter was the number of bays on a primary truss span and the spacing of bays for the secondary trusses. This allowed a straightforward formulation as an optimisation problem with two variables: one continuous variable (bay spacing, between 2 m and 4 m) and an integer variable (number of bays per primary truss span, between one and four). This two-dimensional design space is represented in Figure 5. A parametric system was set up to define the problem geometrically; this allowed logical decisions to be encoded regarding design aspects of the truss, which would vary with the parameters, as well as generating a flexible automated model. One example of the in-built logic can be seen at the tips of the secondary trusses, which either terminate as points or beams depending on the minimum practical truss depth.

An optimisation based on a genetic algorithm was employed to generate the geometry of the trusses and then perform structural analysis under self-weight and wind loading conditions. The multi-objectives of weight and maximum deflection where chosen as fitness measures to be minimised. The final design chosen possessed the lowest weight for the truss structure, satisfying the allowable serviceability deflection limit and taking into account all possible combinations of main and secondary truss dimensions.

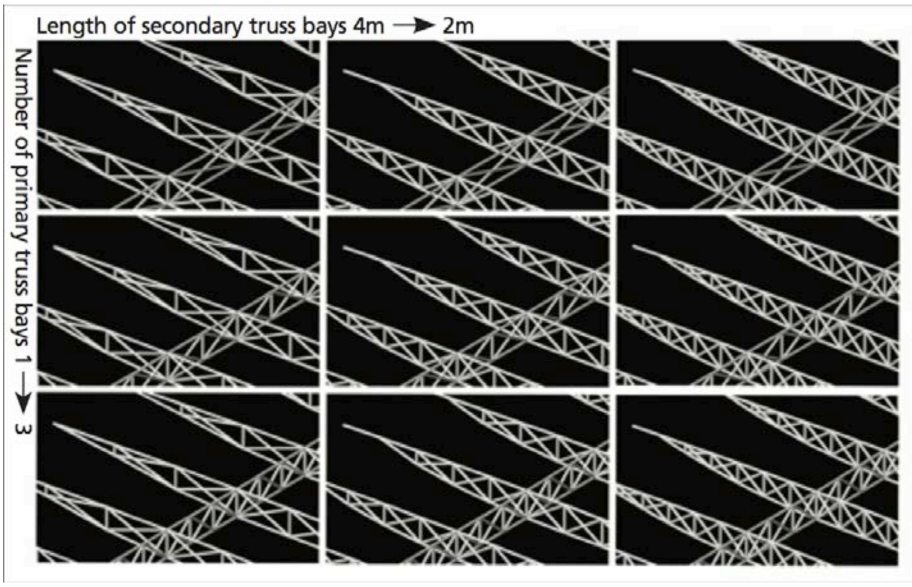


Figure 5: A two dimensional mapping of examples of the two parameters of the model along the vertical and horizontal axis.

Application: low-carbon housing design

The second case study concerns a residential project in Scotland subject to stringent carbon dioxide emissions and financial requirements. The development consisted of 1500 dwellings on a south-facing rural site. A mix of dwelling sizes and types and the layout of these based on architectural considerations provided a fixed development plan.

A key requirement was a 60\% improvement over the carbon dioxide emissions target set by building regulations. There was also a specific limit on the energy use for space heating to avoid solutions that had high energy use counterbalanced by high renewable energy provision. The developer obviously wished to meet the targets in as cost-effective a manner as possible and, in addition, it was necessary to ensure that there was not an excessive risk of overheating in the summer.

This problem was formulated as an optimisation of two objectives – carbon dioxide emissions and cost. The carbon dioxide objective was the percentage by which the dwelling emission rate exceeded the target emissions rate. Carbon dioxide emissions were evaluated using the Standard Assessment Procedure (SAP), the methodology used in England and Wales for building regulations compliance for domestic buildings (DECC, 2011). The cost objective combined capital cost with running costs over 20 years, and was based on data from the cost consultant to ensure it was appropriate to the project. Finally, constraints were imposed to ensure that the overheating risk for all designs was low or moderate and that the space heating requirement was met; both were calculated by SAPs.

The optimisation algorithm used was NGSA-II (Deb et al., 2002), one of the most popular multi-objective genetic algorithms. This was implemented in VBA for Microsoft Excel to facilitate interaction with the SAP calculations in Excel.

Seven variables were included in the optimisation, each taking a discrete value from a predetermined range; all other parameters were set to constant values. The variables chosen addressed: fabric properties (areas of glazing, insulation and air-tightness), heating system (selected from four options – gas, air-source heat pump, solid fuel burner, community biomass), renewable energy provision by way of photovoltaics and solar thermal hot water.

The results presented give an example of an optimal set of designs; these are only valid for the context used here, as defined by the constants used for all other parameters in the methodology.

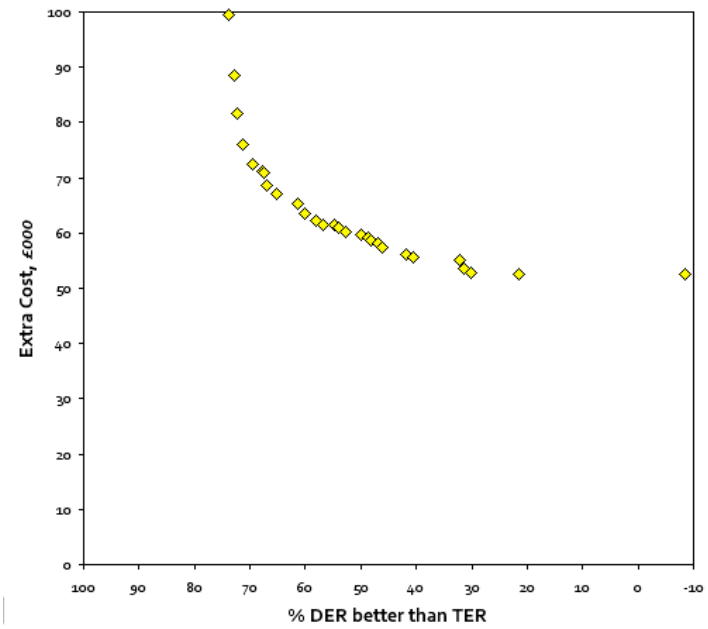


Figure 6: Trade-off front for low-carbon housing problem.

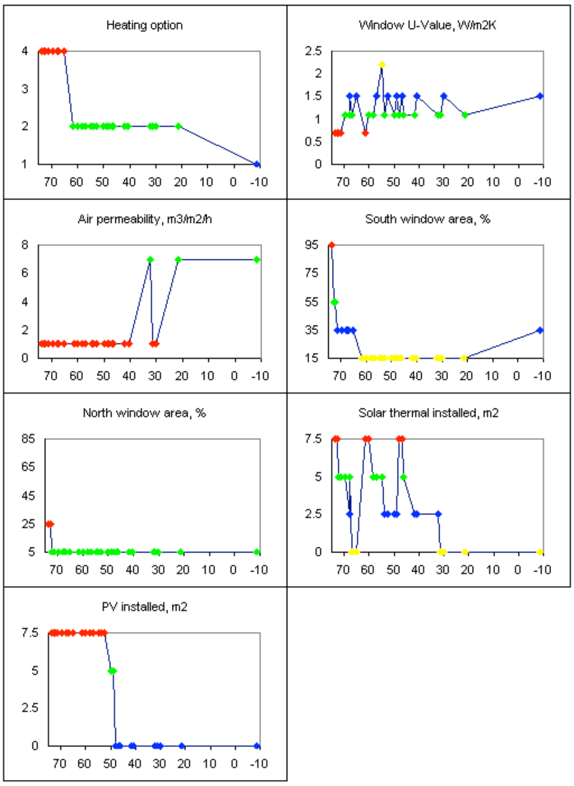


Figure7: Variations amongst optimal solutions for low-carbon housing problem.
The x-axis is the % DER improvement over TER, and the y-axis gives the variable value.

Figure 6 shows the trade-off front for the problem, from 'expensive and low carbon dioxide' to 'cheap and high carbon dioxide' (left to right). Figure 7 shows plots of each variable along the trade-off front, illustrating the nature of the trade-off front solutions. For example the heating system (Figure 7(a)) forms discrete sections, from gas to air-source heat pump to community biomass (as cost increases and emissions decrease); the solid fuel burner option does not appear, so is never an optimal design. With respect to photovoltaics (Figure 7(g)), none are required for up to a 50\% carbon dioxide improvement; this then rapidly steps up to 7·5 m2 per dwelling, the maximum allowable. Where periodic changes appear (e.g. window U-value and solar thermal), performance is improved until a change elsewhere allows the specification to be backed-off again.

Issues to be resolved

The examples in Section 4 demonstrate that it is now practical to perform automated optimisation on certain aspects of design problems. However, these methods are not a replacement for designers: the approaches still require an underlying system or model to optimise. Nevertheless, designers are now able to introduce a greater level of flexibility where appropriate and allow optimisation methods to perform the evaluations. This does, however, require designers to fully understand what they desire as an outcome so that they can correctly formulate the problem. It is common to perform multiple optimisations to answer different facets of the same problem. The exploratory and questioning nature of the designer is thus still at a premium even in this automated process.

The computation time required for the models to run is not trivial. Whereas simple rules of thumb can be introduced to allow quick appraisals, large run times may be required for detailed structural, thermal or fluid simulations. Poor communication methods between different programs can also limit the level of automation that is possible. These form the main limitations to these techniques: it may only be possible to apply them to simplified sub-sets of the overall problem or it may be necessary to use simplified analysis methods available in packages with good interoperability.

Conclusions

The approaches outlined in this paper have found wide application in industries other than civil engineering. With the introduction of readily accessible tools for creating parametric design models as well as the emergence of standard multi-objective optimisation algorithms, barriers for adoption in the construction industry have been significantly reduced. Benefits over traditional methods include greatly reduced time per design option trialled (countered by increased set-up time), improvements in performance for complex problems and increased rigour in the design process. These advantages will be most significant on projects that push the boundaries of performance – and hence small improvements are important – or on projects with high repeatability – and thus savings are multiplied.

The range of uses of the techniques means it is difficult to generalise regarding their role in the design process. They can be used when the complex nature of the problem demands the use of advanced methods – and sufficient time and resources are available. They may be used when it is possible to abstract meaningful simplifications of a problem, for example examining a typical zone of a larger building. Alternatively, they may be used to examine many general problem cases, to develop design rules and guidelines that are then applied at project level.

Further information regarding multi-objective optimisation is available from conferences proceedings (e.g. the ACM Genetic and Evolutionary Computation Conferences, IEEE Congress on Evolutionary Computation, International Conference on Evolutionary Multi-Criterion Optimization), journals (e.g. IEEE Transactions on Evolutionary Computation, Journal of Multi-Criteria Decision Analysis) and the internet (www.lania.mx/~ccoello/ (evolutionary multi-objective optimisation repository), www.iitk.ac.in/kangal/ codes.shtml (NSGA-II C code)).

Acknowledgements

Funding for this research has been provided by EPSRC and Buro Happold Ltd. Both authors are research engineers with Buro Happold and with the Industrial Doctorate Centre in Systems, Universities of Bristol and Bath, UK.

References

Coenders, J.L., 2009. Exception as a rule in computational design. In *Evolution and Trends in Design, Analysis and Construction of Shell and Spatial Structures*. International Association of Shell and Spatial Structures. Available at: http://dspace.upv.es/manakin/handle/10251/6868 [Accessed March 1, 2011].

Deb, K. et al., 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. *Evolutionary Computation, IEEE Transactions on*, 6(2), pp.182-197.

Fong, K., Hanby, V. & Chow, T., 2006. HVAC system optimization for energy management by evolutionary programming. *Energy and Buildings*, 38.

Hesselgren, L., Charitou, R. & Dritsas, S., 2007. The Bishopsgate Tower Case Study. *International Journal of Architectural Computing*, 5(1). Available at: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.112.3981 [Accessed March 1, 2011].

Huang, W. & Lam, H., 1997. Using genetic algorithms to optimize controller parameters for HVAC systems. *Energy and Buildings*, 26.

Koumousis, V. & Georgiou, P., 1994. Genetic Algorithms in Discrete Optimization of Steel Truss Roofs. *Journal of Computing in Civil Engineering*, 8(3).

Marks, W., 1997. Multicriteria optimisation of shape of energy-saving buildings. *Building and Environment*, 32(4), pp.331-339.

Ren, Z. et al., 2011. Multi-disciplinary collaborative building design--A comparative study between multi-agent systems and multi-disciplinary optimisation approaches. *Automation in Construction*, In Press, Corrected Proof. Available at: http://www.sciencedirect.com/science/article/B6V20-51YXM50-1/2/27d497b4ce5e8e4495b6180c3a77a242 [Accessed February 28, 2011].

Roy, R., Hinduja, S. & Teti, R., 2008. Recent advances in engineering design optimisation: Challenges and future trends. *CIRP Annals - Manufacturing Technology*, 57.

Shepherd, P. & Hudson, R., 2007. Parametric definition of Lansdowne road stadium. In *Proceedings International Association of Shell and Spatial Structures*.

Wang, W., Zmeureanu, R. & Rivard, H., 2005. Applying multi-objective genetic algorithms in green building design optimization. *Building and Environment*, 40.

Williams, C.J.K., 2001. The analytic and numerical definition of the geometry of the British Museum Great Court Roof. In *Mathematics & Design*. Available at: http://opus.bath.ac.uk/14111/ [Accessed March 1, 2011].

Zolfaghari, A.R., Heath, A.C. & McCombie, P.F., 2005. Simple genetic algorithm search for critical non-circular failure surface in slope stability analysis. *Computers and Geotechnics*, 32(3), pp.139-152.

Towards Ubiquitous Structural Frame Design Tools

Sam Conrad JOYCE
Research Engineer/Analyst
Buro Happold SMART
London, UK
Sam.Joyce@BuroHappold.com

Al FISHER
Senior Analyst
Buro Happold SMART
Bath, UK
Al.Fisher@BuroHappold.com

Chris WILLIAMS
Senior Lecturer
Bath University, Bath, UK
C.J.K.Williams@bath.ac.uk

Shrikant SHARMA
Associate Director,
Buro Happold SMART
Bath, UK
Shrikant.Sharma@BuroHappold.com

Summary
A collection of methods and algorithms are presented which aim to aid in the automatic designing of steel structural frames, under both deflection and stress criteria. A virtual work method is utilised assessing local contributions of members to global nodal deflections. It is the invention of the authors to apply these methods iteratively to indeterminate structures. Complementary algorithms which design members for stress have also been devised following Evolutionary Structural Optimisation principles. In this way an optimisation suite has been developed which can be used systematically to design structural frames. It has been found practically that this method converges on a low weight solution, with designs found much more expediently in comparison to typical beam selection methods. The implementation of these advancements in the form of useable tools for application by non-specialists is discussed. Finally possible directions for further work are proposed.

Keywords: *Optimisation; design tools; structural frames; member design*

1. Introduction

1.1 The problem

Design of structural frames has always been a mainstay of engineering firms since Gustav Eiffel. As the size of our population increases so do our cities resulting in a rise in the number of large buildings requiring design. It is thus pertinent to have tools that can aid in this design and engineering process. Away from more involved holistic design optimisation methods [1] there is a requirement to minimise the amount of material used by such structures. The advent of Finite Element programs for the calculation of structural frames has reduced the analytical workload for engineers immensely, and this has been levered by most in the industry. These automated methods of design however do have the restrictive constraints of forcing definition of members in a generic way; a standard example is grouping of top and bottom chords of

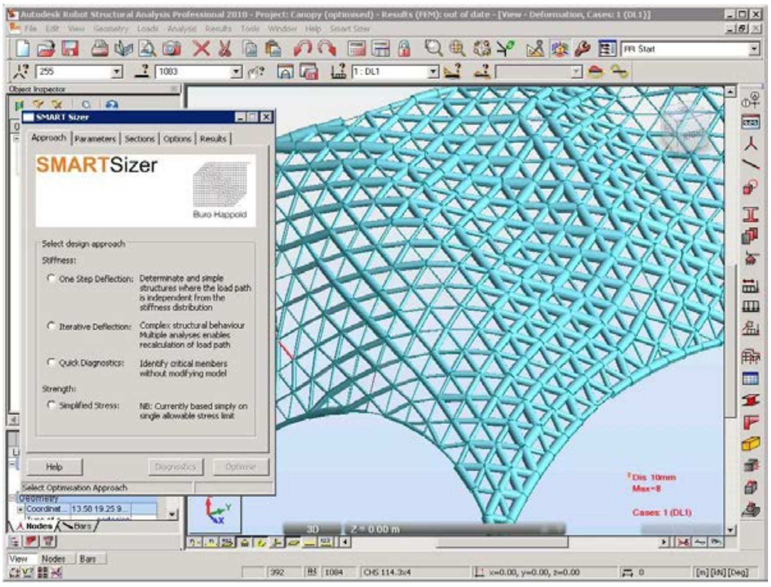


Fig. 1: Image showing tool in analysis environment alongside an optimised structure

space frames or trusses into uniform sections. In this way automation has reinforced the traditional approach of designing for worst case with greater levels of element repetition. This is less to do with being unable to size all elements individually, but more a reluctance to do so on the part of engineers. This is brought about by the practicalities of available design time relative to the speed of the numerical analysis, which can now be completely automated and available at the touch of a button. It can be shown from theory such as Mitchell [2] and built projects such as the British Museum Great Court Roof [3] that material economies can be gained from mass variation of member sizes without being an overly intensive fabrication process especially with the advent of CAM cutting systems [4]. However until now this has simply not been possible in most practical time frames and work flows.

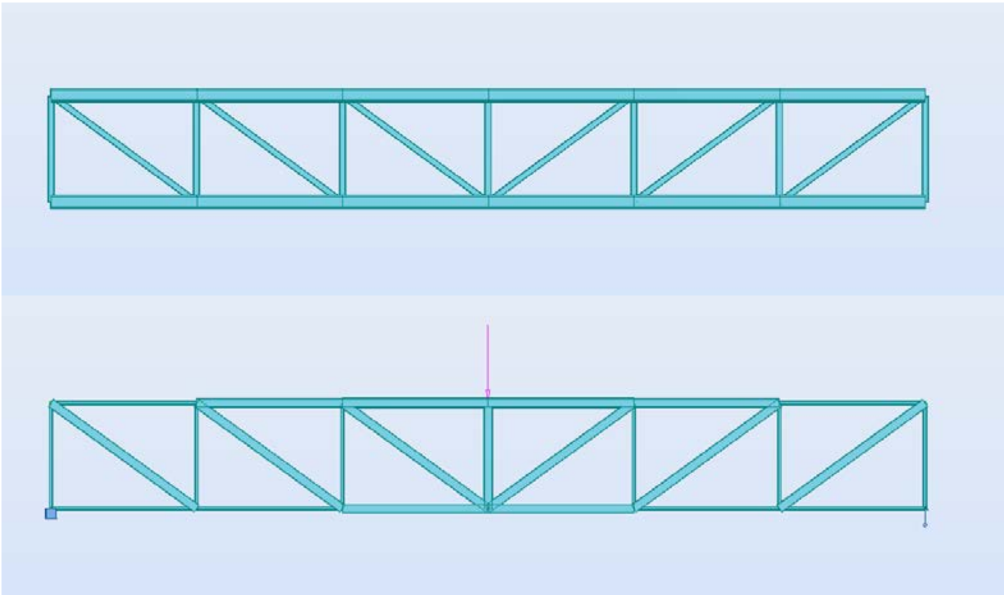


Fig 2: Image showing typical section grouping for a simply supported truss loaded in the centre, below a more optimal design with a change in elements along chord length

1.2 Our Overall Aim

It was the desire of the authors to develop a process in structural design which made use of the autonomy and speed that computers exhibit in FE analysis, but towards effective member design. Our aim was to have a tool that could respond to the two main design criteria;

1. Serviceability limit state measured by maximum allowable deflections from key points on the structure (mid spans, inter-story drift, etc.), which was the member design for stiffness.
2. Ultimate limit state design where all components of the structure are not allowed to exceed a given stress criterion.

The scope for this phase of the tool development was defined by identifying generic processes would be practically and widely useful and could be achieved in a realistic timescale. In addition it was decided that this process should not be intrusive on the geometry and topography of the structure; simply being a member sizing tool. In this way the tool would integrate much more readily into everyday workflows as it was found that sizing is a discrete task in design whilst geometry and topology is rarely modified without serious and lengthy consultation between design team members.

2. Optimisation algorithm

2.1 Context

Methods previously implemented by SMART Solutions and University of Cambridge [5] have ranged as widely as basic one-step methods to advanced genetic algorithms, it was the desire of the authors to present a method that could be understood and trusted by the majority of design engineers. So that the methods developed and used on complex projects could be applied more widely across engineering practice.

Current optimisation work in the field of structural optimisation can be categorised as two main areas. Firstly 'traditional' approaches solving heavily constrained structural problems with linear optimisation techniques [6]. Secondly techniques commonly called "evolutionary structural optimisation" ESO, operating on open design domains with the algorithm able to introduce material anywhere within a fixed domain [7] [8] [9]. Whilst these methods offer an approach which could provide the best solution given a completely open design, it is the experience of the authors that practically this is rarely applicable where projects have many additional aesthetic and construction constraints limiting complexity [10]. This is something that the inventors of such methods have also highlighted and there are also attempts at reverse engineering these free form shapes back into conventional steel sections [11].

It was highlighted that the most common problem confronting engineers currently was a fixed frame topology and nodal geometry with desired element section types (SHS, UB etc). The main job of the engineer in this case is to size the members for minimum weight [12]. The tool is to be seen initially as useful during rapid assessment under the governing analysis criteria, which is stress and serviceability that define the section properties. The more detailed criteria such global buckling or dynamics are to be integrated later.

2.2 Iteration for Stress Criterion

```
Set frame including member sections
Until numloops>max numLoops
      Apply loads
      Analyse

      For each beam in frame
            Find max stress in beam = Smax
            If (Smax- Sideal) > tolerance
                  Increase cross-sec area An
                  An+1 = An*factor
                  Find   smallest   beam   with
Area>=An+1 in section data base
            End if

            If (Sideal- Smax) > tolerance
                  Decrease cross-sec area An
                  An+1 = An/factor
                  Find   smallest   beam   with
Area >= An+1 in section data base
            End if
      Next beam

      If no beam size changes
            Converged
            Break loop
      End if
Next loop
```

The approach employed for stress solution is very similar in concept to ESO type techniques. Given a predefined loading condition, it follows that by increasing material in over-stressed areas and decreasing material in areas that are under-stressed that the structure's global efficiency or measure of utilisation will improve. This assumes that the loading introduced by self weight is small compared to the governing load to assure convergence. To ensure convergence, this method calls for a degree of tolerance to be applied to the target utilisation to prevent the member sections oscillating between section sizes. An example pseudo-code algorithm is shown above.

2.3 Deflection Criterion

Initially the work of Baker [13] generated the theoretical foundation of the authors' applied technique. In summary this method applies a unit load in the direction of the considered deflection (multiple constraints can additionally be applied to multiple nodes). Using this unit load each member's contribution to the deflection is calculated. For the allowable deflection specified for a specific load-case the members are sized in proportion to their contribution to give the optimum member sizes.

The failing of this method is in its inability to define exact member sizes for indeterminate structures. Here the load path is not invariant to the assignment of section sizes. This prevents constant member contributions from being calculated in a single step as they are dependent on the relative stiffness between members along competing load paths. Thus it is not possible to use this method in a one setup manner for the majority of structures a designer is confronted with. Redundancy is key in the design of real structures and as such analysis of indeterminate structures with complex load path behaviour is critical.

As an alternative, here a method is presented which allows for an iterative approach to member design. In this case the design is analysed and the member contributions calculated based on the initial member sizes. Then a percentage 'α' of the most contributing elements is sized up by a fraction 'β' of their current cross-sectional area. Typically α and β are between 10-50% and 1.1-1.5 respectively. Crucially the structure is then re-analysed and the member contributions reassessed. By iterating this process a load path can be developed increasing the capacity of the pertinent parts of a structure.

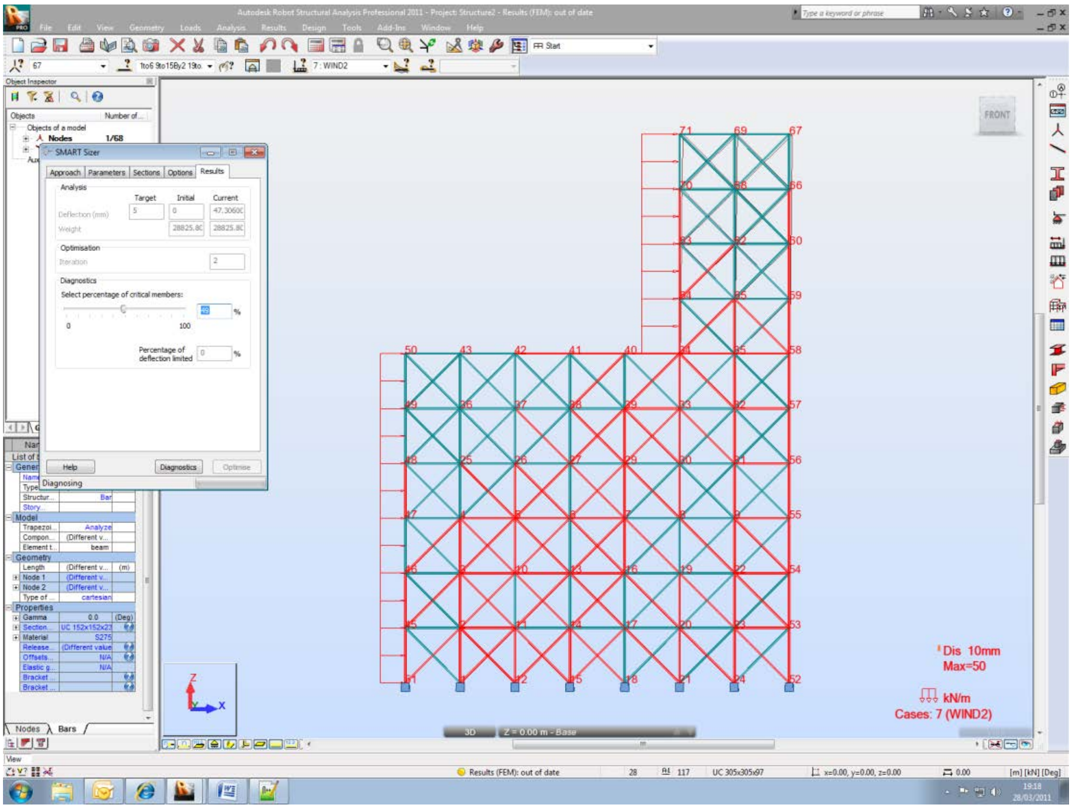


Fig 3: Image showing the top 50% contributing members coloured in red for the deflection of node 67 in the horizontal direction for the load case shown.

3. Integration of Optimisation Techniques

For the successful application of tools into a working environment it is important that they are capable of complementing each other. It was decided that the optimisation should be piecemeal (both stiffness and stress) to allow autonomy of procedure for the common conditions that there is a dominating design case and thus a complete optimisation is not required.

To achieve a coupled stress and deflection based optimisation a multi-phase process of successive granularity is proposed. Ultimate limit state and serviceability requirements both must of course be satisfied. However the final solution reached is heavily dependant on the approach taken and the order these requirements are considered. This approach starts from a global view of the structure's stiffness distribution and behaviour before considering each and every member for their local stress requirements. This global then local approach to design helps to improve the stress based design as satisfaction of the deflection criterion enforces a global load path which the stress based optimisation can then tune to ensure there is no over utilisation.

This also requires the possible allowance of the tool to only size up the member sizes during each phase. In other words for a multi-phase optimisation, members with low stress utilisation during later phases may have been sized up to perform a key role in controlling the global stiffness distribution and thus should not necessarily be sized down. Running the optimisation in this order, from macro- to micro-constraints, it is possible to superimpose the two solutions, satisfying multiple design objectives, without considerable over sizing of the design.

The selection of the dominating load case is also important, such that the optimisation can be run in order of reducing importance too. For multiple load cases the design is then optimised for the governing load case(s) with the later runs become more of an automated checking procedure. It is possible to see the dominating load case easily using these techniques by running an optimisation for each on the same initial low weight design and seeing which is heavier. A considerably heavier 'optimised' design for one case will show the dominating case.

4. Implementation

There is still much to be improved in conventional structural engineering packages before they become practical for expedient implementation of computational design processes. There exists a big differentiation between relatively academic platforms for structural analysis such as Matlab/Ansys which provide easy automated interface, and the more typical professional design engineering tools such as Robot/SOFiSTiK. The professional tools do much more to present an intuitive user interface but this is at the expense of ease of automation.

However for practical use in a design office these optimisation capabilities should be available on the most commonly used platforms. For this reason our tool was developed as a plug-in for Autodesk Robot Structural Analysis Professional. To make this work sufficiently fast it was necessary to automate Robot directly from a lower level Dynamic Link Library (dll) program in-process with Robot rather than the easier to implement interfaces often used for remotely automating programs such as VBA. For practical use of the tool in live design projects user experience has been a key driver. The program was developed to run in process with Robot was to enable an acceptable speed for the tool to be achieved. The speed of running the optimisation tool on a model is obviously dependent of the size of the model; however for a relatively simple model (less than 300 elements) the desired running time of less than a minute was achieved.

5. Use, projects, feedback

Much of this work was developed alongside major large-scale steel design work; one such project being the Louvre Abu Dhabi [14]. When applying research approaches to design projects, important issues of practically become quickly relevant. Designers required extra control over the member types and this control was mirrored by the tool's selection of alternative sections. It was found that functionality was required to allow engineers to make most use of the tool by allowing sections only to be switched with sections of the same type, whether this was user defined (allowing for radical changes of geometry) or the standard steel section groups available on the section databases (such SHS, UB etc). By allowing section changes to sections with the same name within custom databases it is possible to have a set of available sections limited only by the number added by the individual. It is the desire of the authors to introduce parametric generative sections in the future.

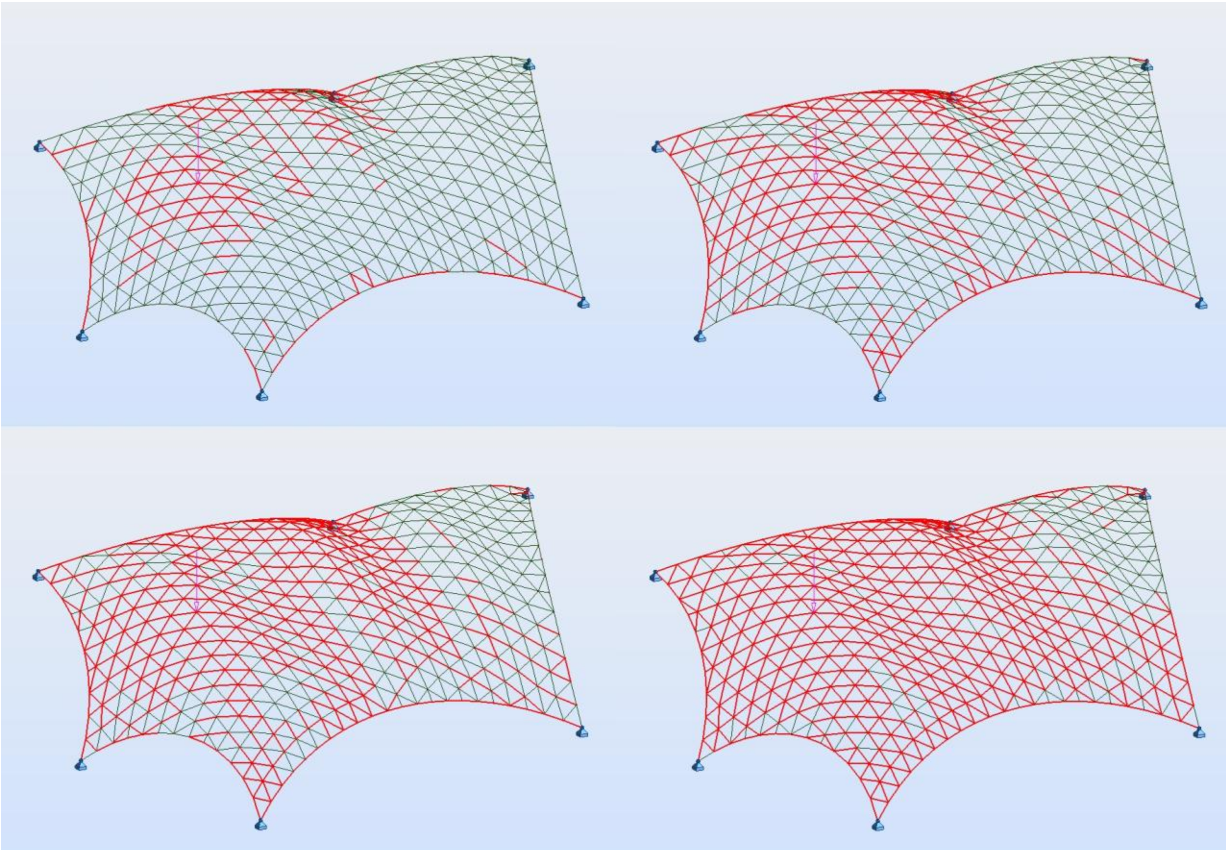


Fig 4; Showing the members contributing most to the vertical deflection of a point loaded vertically on the leftmost peak of the structure with the purple arrow. Images show 20%, 40%, 60% and 80% highest contributing members coloured in red.

It was also found that there was much benefit in understanding of what the tool did by offering a “diagnostics” mode where the member contributions for a load case could be highlighted see fig.4. By allowing dynamic modification of the percentage of members chosen, engineers are able to gain an understanding of the key members contributing to deflection at that state. This was seen as a benefit as an enabler of understanding of the structural behaviour of the design as well as the optimisation process. Transparency is an important feature of the tool where traditional optimisation approaches can often be seen as unintuitive black boxes.

6. Further Work

As explained the techniques and algorithms developed are a direct product of application to real world projects. As such there is a balance between a reliable, dependable tool and advancing the speed and capability. Some advancements to this method currently being developed are discussed below.

Whilst there is considerable scope for a more advanced combined method of analysis taking into consideration all load cases, multiple load cases are currently optimised using a sequential approach. A combined load case optimisation has been investigated. In this case the highest maximum absolute stress utilisation or deflection contribution is taken for each element, with the algorithm sizing the elements as before.

A member sizing procedure is implemented where the increase in size of the element is proportional to the utilisation of the beam;

$$A_{n+1} = A_n + A_n \cdot \beta \cdot (U_n - U_{opt}) \qquad (1)$$

Where 'U_n' is current utilization and 'U_{opt}' is the optimal utilization and 'β' performs the same function as in section 2.3. In this case large deviations from the utilization rise faster than small deviations and it is believed that the convergence speed will be improved. This method is best suited for truss structures, for structures where the stresses are not driven by the cross-sectional area, more work needs to be done to augment the member size in a more efficient way depending on the nature of the force causing the stress, for example improving the second moment of area in members with high bending stresses. This improvement would be effective on improving the solution of both stress and deflection conditions.

7. Conclusion

A procedure for automatically sizing members has been presented. This allows for a much higher level of detail of member design to be implemented at an early stage by the use of a suite of optimisation tools. The application of automated member sizing has great capability for reducing the work load of the design engineer as computational finite element analysis had previously. Like all automations there is a balance to be struck between the level of control that the computer and user have. In this case the overall procedure has been designed to be followed and understood by the engineer with the computer taking up the piecemeal but labour intensive sections.

The authors are of the opinion that a greater provision for practical automation of structural analysis software from programming type environments is required, rather than the closed system mindset that seems prevalent in those developing these programs.

Finally it is also important to factor in the requirements of the user when designing these techniques to make them understandable and approachable so that they may be widely used and thus have maximum impact.

8. Acknowledgments

Funding for this research has been provided by EPSRC and Buro Happold Ltd. Sam Joyce is a research engineer with Buro Happold and with the Industrial Doctorate Centre in Systems, Universities of Bristol and Bath, UK.

9. References

[1] EVINS R., JOYCE S. C., "Multi-objective optimisation: higher 'performance' for lower 'cost'', More *for Less,* Civil Engineering Journal, November, 2011, (in press).

[2] MICHELL, A. 1904. "The limits of economy of material in frame structures", Philosophical Magazine, Vol. 8, 1904, p. 589–597

[3] BROWN S., "Engineering the British Museum Great Court Roof", *Wide span Roof Structures*, Thomas Telford, London, 2000, pp. 283-286.

[4] SISCHKA J., "Engineering the Construction of the Great Court roof for the British museum", *Widespan Roof Structures*, Thomas Telford, London, 2000, pp. 199-207.

[5] WINSLOW P, FISHER A, SHARMA S., "Design tools for structural optimization", *Evolution and Trends in Design, Analysis and Construction of Shell and Spatial Structures*,

Proceedings IASS, 2009, pp. 1175-1186.

[6] Raphael T. Haftka, Zafer Gürdal, Elements of structural optimization, Kluwer, Netherlands, 1992, pp 71-113.

[7] XIE Y.M., Huang X., Tang J.W. and Felicetti P. "Recent Advances in Evolutionary Structural Optimization", Keynote Lecture, *Frontiers of Computational Science Symposium*, Nagoya University, Japan, 2005.

[8] SASAKI M., Morphogenesis of Flux Structure, AA Publications, London, 2007

[9] ALLAIRE G., JOUVE F., and TOADER A., "Structural optimization using sensitivity analysis and a level-set method", *Journal of Computational Physics*, Vol. 194, Issue. 1, 2004, pp. 363-393.

[10] BURRY J., BURRY M., *"Qatar Education City Convention Centre"*, The New Mathematics of Architecture, Thames and Hudson, London, 2010, pp. 130-133.

[11] LIANG Q. Q., XIE, Y. M., and STEVEN G., *"*Optimal topology design of bracing systems for multi-story steel frames". *Journal of Structural Engineering*, Vol. 126, Issue. 7, pp. 823-829.

[12] EPP L., BERRY K., HART R., "Cairo Expo City – a Free-form Spatial Roof Structure", *IASSE-IASS Symposium*, London, UK, September 20-23, 2011.

[13] BAKER W.F., "Stiffness optimisation methods for lateral systems of buildings: A theoretical basis", Electronic Computation, 1999, pp269-278.

[14] SHRUBSHALL C., FISHER A., "The practical application of structural optimisation in the design of the Louvre Abu Dhabi", *IASSE-IASS Symposium*, London, UK, September 20-23, 2011.

Appendix B

Case Studies Overview

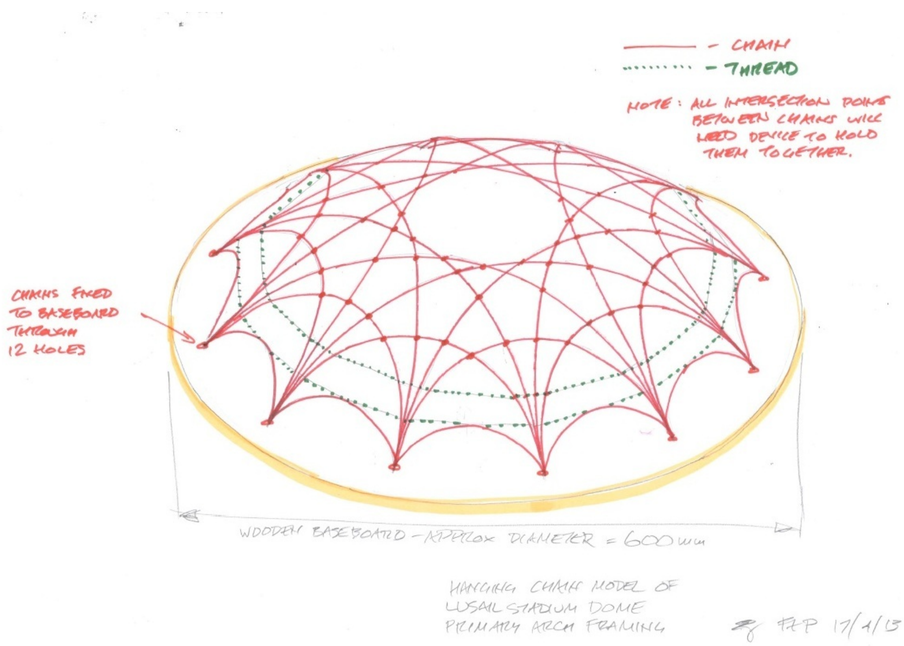


Figure B-1: Sketch of intent for the hanging chain model. Source Foster + Partners.

B.1 2022 World Cup Main Stadium Qatar [1863/1864]

- Type: Stadium
- Location: Lusail, Qatar
- Size: Site: 700,000m^2
- Collaborators:
  - Internal: Studio 3+1, Modelshop
- Personal involvement:
  - Stage: concept
  - Duration: 2 days
  - Software: Rhino Grasshopper

Description: Stadium bowl is covered by a large dome made of arches which lay on the intersection between an analytical surface (ellipsoid and later an elliptic paraboloid) and the arch paths in plan.

Involvement: A study was undertaken to create a physical hanging chain model to visualise and understand the shape of the roof if derived in this way. A computational dynamically relaxed model (using custom basic dynamic relaxation) was generated intended to fit this as closely as possible. Data output to provide chain lengths and node positions so that the model-shop could produce

Research Value: Parametric implementation of form optimisation and feedback by architects on requirements and issues for effective control of these physical systems in a design rather than analysis context. Comparisons of the speed of creation between modern day physical modelling process and computational equivalent in favour of the computational approach.

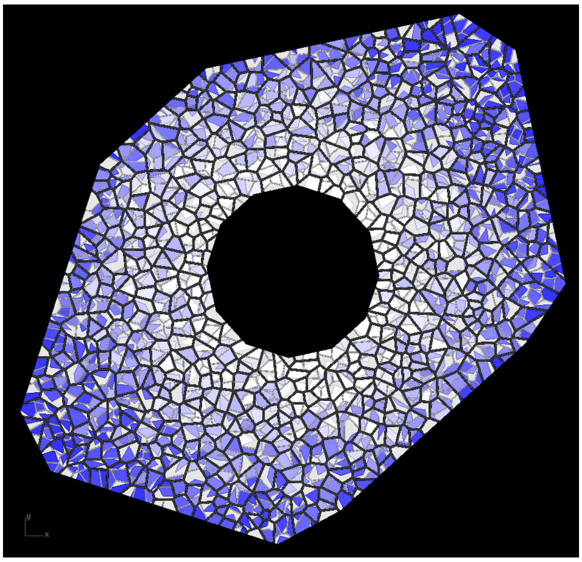


Figure B-2: Plan image of rug design. Source Foster + Partners.

B.2 SAMBA Tower Rug Design[1806]

- Type: Interior
- Location: Riyadh, Saudi Arabia
- Size: $100m^2$
- Collaborators:
  - Internal: Interior Design, Studio 1
- Personal involvement:
  - Stage: Detail
  - Duration: 3 days
  - Software: Rhino Grasshopper

Description: Design development for a custom woven rug for the main executive meeting area.

Involvement: Creation of a purely aesthetic geometrical pattern to progress the clients desire for a unique geometric but not symmetrical rug design, which incorporated a gradient of both colour and pattern density.

Research Value: Investigation into the use of parametric concepts purely for unbounded aesthetic requirements. Useful to contrast with more constrained engineering centric design problems.

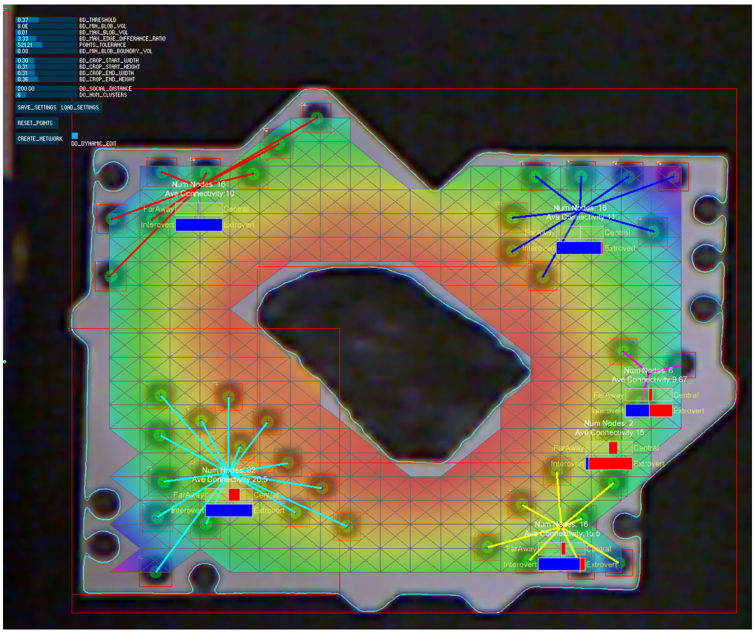


Figure B-3: Screen shot of tracking table, overlaid with office metrics

B.3 Work Space Tool [1889(Shaw)]

- Type: Work Place Design
- Location: Office
- Size: 10-1000m^2
- Collaborators:
  - Internal: Work Place Consultancy
- Personal involvement:
  - Stage: Concept and Fit-Out
  - Duration: 3-4 weeks
  - Software: Processing, Image-Processing

Description: Speculative development of a system to aid the design team in the effective configuration of office floor plates, based on relevant metrics.

Involvement: Created a standalone system able to either take in floor plates and allow for mouse based dynamic arrangement of the seating or a live feed version that identifies floor plates and seating markers. This then shows data about the spatial and social connectivity of the arrangements, including a K-Means based clustering method to identify potential working zones and their internal and external connectivity. A further extension was undertaken to move the seating positions based on social forces, to at as a local optimiser of existing proposals.

Research Value: Development and critical assessment of analytical machine learning methods applied to work place consultancy. Focused feedback from design team on requirements for a client centric interface. Investigation into alternative interfaces to traditional input mechanisms via camera and image detection highlighting the relative difficultly in set-up and implementation which is less practical in many situations. Identification of the importance of clear data visualisation both for the introduction of new techniques to domain specialists (workspace designers), but also non designer clients to enable tools can have impact in decision making.

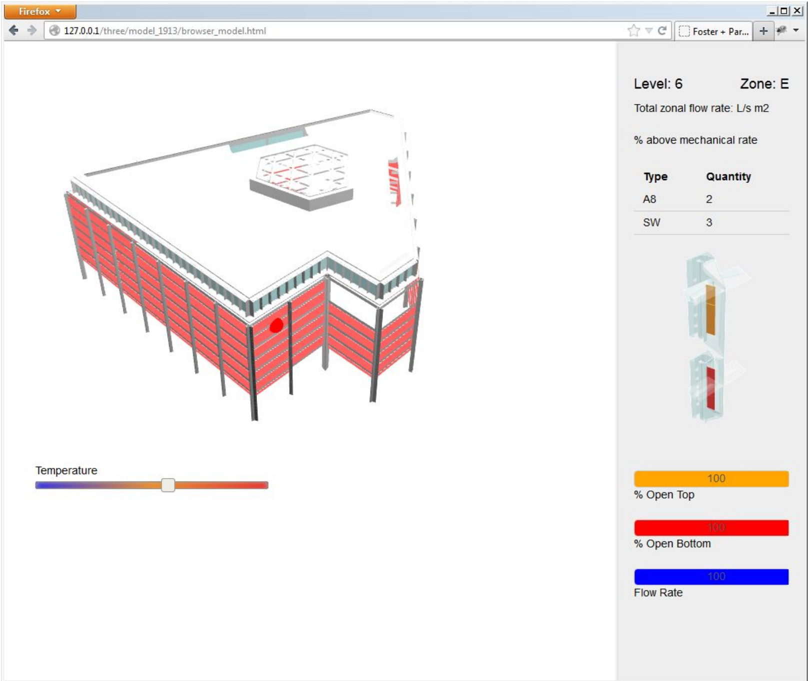


Figure B-4: Screen capture of the web interface

B.4 Bloomberg [1942]

- Type: Office Development
- Location: London, England
- Size: 46,500m^2 GFA
- Collaborators:
  - Internal: Engineering 2 (environmental)
- Personal involvement:
  - Stage: Detail
  - Duration: 2 weeks
  - Software: Web Development; HTML, Java Script, d3.js, three.js

Description: Large scale office development scheme with complex natural ventilation strategy. It was believed that by producing an interactive app that would simplify explanation of the way the system would work to the client and potential building managers.

Involvement: All of the environmental logic to determine the state of the natural ventilation system was transferred from the engineering teams spread sheets to Java Script to enable it to be linked to a interactive 3D model. This allowed for the positioning of the openings to be controlled by the temperature slider, with overall state display on the building and a close up of the potion of specific ventilation components shown.

Research Value: Investigation into the creation and value of complex 3D models on web based platforms with feedback by environmental engineers on the usefulness of this in their work flows. Study showed the strength and ease of access that web platforms afforded. Work highlighted the relative complexity of implementing excel based logic into JavaScript to run on the browser, favouring a 'pre-baked' approach to generating data separating analysis logic and visualisation logic.

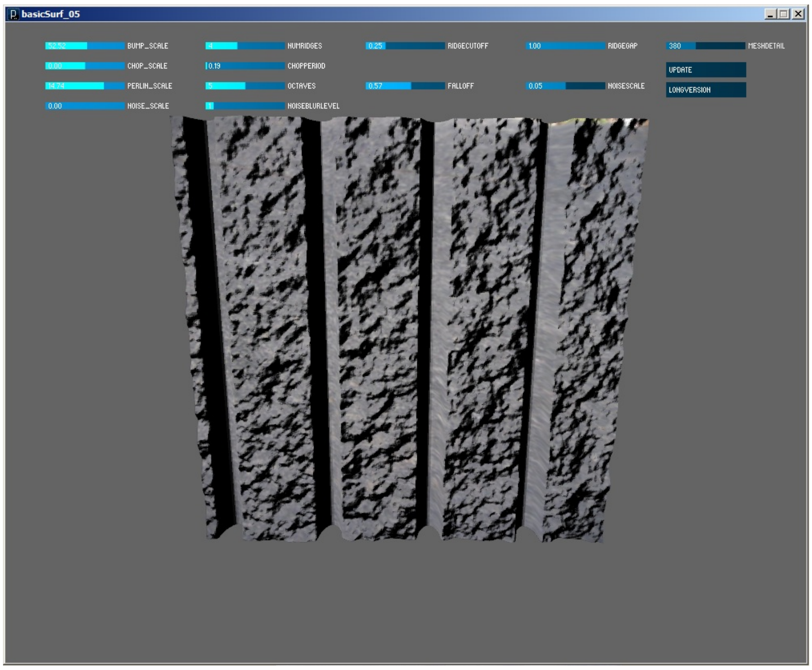


Figure B-5: Screen capture of stone design interface

B.5 Shanghai Bund [1960]

- Type: Retail + Hotel + Office
- Location: Shanghai, China
- Size: 426,000m^2 GFA
- Collaborators:
  - Internal: Studio 5
- Personal involvement:
  - Stage: Scheme
  - Duration: 3 days
  - Software: Processing

Description: Large mixed use development using rusticated stone faade ’frames’ as the main architectural articulation. The definition and visualisation of these faades was hard to achieve so computational support was required.

Involvement: Developed a standalone tool that utilised Gaussian and Purlin noise to mimic the rusticated stone textures. This was extended to incorporate control over bands of smooth ridges in the pattern. The tool gave its own visualisation of the results as they where changed in real time. These could then be exported in large non-repeating sections for high quality rendering.

Research Value: Investigation allowed for assessment of completely bespoke one off tool creation for iterated tasks. Study showed increased exploration of options enabled by the speed of generation by the tool leading to more design refinement as compared to the original manual methods.

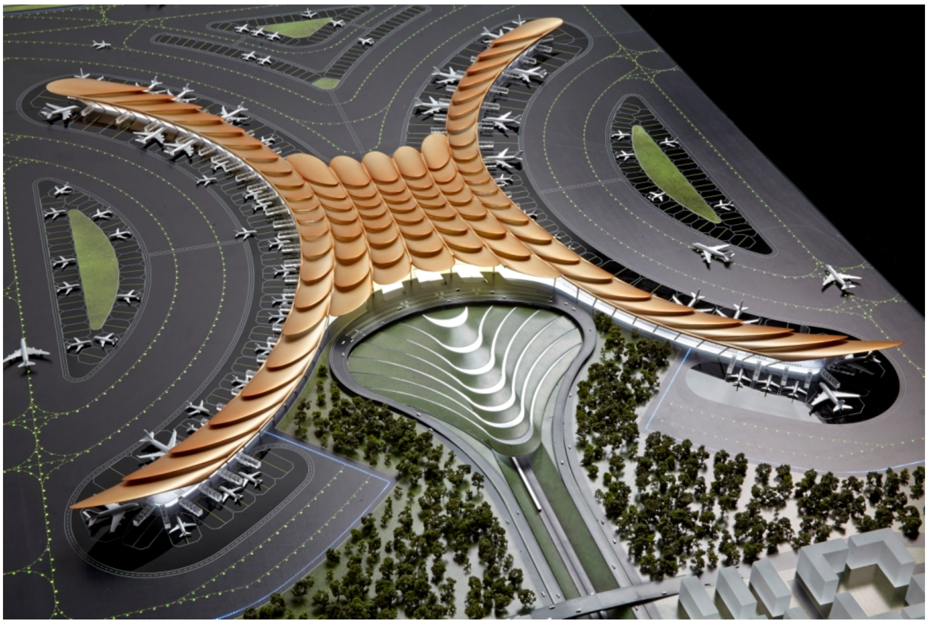


Figure B-6: Photo of the presentation model. Source Foster + Partners.

B.6 Beijing South Airport [2012]

- Type: Airport
- Location: Beijing, China
- Size: Roof Area 380,000m^2
- Collaborators:
  - Internal: CEO, Studio 3, Visualisations, Model Shop
- Personal involvement:
  - Stage: Competition
  - Duration: 1 Month
  - Software: Grasshopper, Oasys GAS, Structural Link

Description: Creation of 'Phoenix' like geometry for a prosed south Beijing airport terminal. Project required the fast development of a visually compelling roof scape which also satisfied environmental, structural, site boundaries, and functional requirements.

Involvement: Managed and created much of the roof geometry taking input from a very small design team including the then C.E.O. of F+P. The grasshopper model was used as a basis for geometric studies on lighting performance, visual column spans. Also this model was used to automatically generate the underlining structure including thin almost shell like space frame structures that make up the 'feathers'. These where automatically passed into GSA from Grasshopper via the F+P structural tools to check approximate structural compliance and efficiency.

Research Value: Case study experience of using a parametric model as the central representation of complex geometry which interfaces with multiple different engineers and stakeholders. Data collected on the interaction between parametric option creation and high-level design decision making. Identification of needs and implementation of computational methods to speed up structural analysis with critical feedback on benefits and failings of these works in short time frames.

Figure B-7: Visualisation of global reach of short-haul and long-haul flights from London with cities sized by estimated metro GDP

B.7 Thames Hub[2033]

- Type: Master Planning
- Location: London + World
- Size: 23,400,000m^2
- Collaborators:
  - Internal: Studio 3
  - External: Hambalt (Strategy Consultants)
- Personal involvement:
  - Stage: Proposal
  - Duration: 2 Months

– Software: Web Development; HTML, Java Script, d3.j

Description: A strategic project to investigate the potential position and role of a new hub airport in London. Requirement was to investigate the global drivers and factors behind the development, and then to explain this complex analysis within a report to government commission tasked to offer recommendations for future aviation development in London.

Involvement: Investigated along with our econometric specialists what where the USP of London as a transport hub as well as its economic and geographical position. Developed interactive web based visualisations of the complex data sets of over 4300 metro areas to explain global connectivity. These tools where later shown to the London Mayor and London Development Agency

Research Value: Large in-depth investigation into the fundamental techniques and current technologies used in data-visualisation. Applied techniques with feedback from both field experts, designers and the public on the ease of interpretation of various developed data views. Identification of the web as a medium to provide deep analysis of complex data whilst still making it available to a wide range of people. Specific research on high dimensional and geographic data sets and use of interactivity to enable greater insight from single visualisation.

Figure B-8: Visualisation of Bangalore Development. Source Foster + Partners.

B.8 Bangalore [2040]

- Type: Residential
- Location: Bangalore, India
- Size: SITE 20,500m^2
- Collaborators:
  - Internal: Studio 1
- Personal involvement:
  - Stage: Concept
  - Duration: 1 month
  - Software: Processing

Description: Residential development planned to be on 20m high columns to improve ventilation and aspect. Positioning of residential 'pixel' units in 3D space undertaken to maximise cross-ventilation and views.

Involvement: Developed a two scale structural optimisation strategy. Utilising a iterative member sizing routine to utilise material usage and ultimate-limit-state compliance, then using a Genetic algorithm to optimise the 'pixel' positioning for a given placement of support cores to minimise the structural material after a given option has been sized using the aforementioned routine.

Figure B-9: Model of Real Madrid FC stadium redevelopment proposal. Source Foster + Partners.

B.9 Madrid Stadium Roof [2104]

- Type: Stadium
- Location: Madrid, Spain
- Size: 44,000m^2
- Collaborators:
  - Internal: Engineering 01 (Structures)
- Personal involvement:
  - Stage: Competition
  - Duration: 2 weeks
  - Software: Rhino, Grasshopper, GSA, Structural Link

Description: Proposal to redevelop Madrid F.C.'s Santiago Bernabu stadium, to improve and enlarge the size without negatively impacting on the sit and its surrounds. Design involved a completely new roof to more fully cover the seating in-line with new FIFA excellence ratings.

Involvement: Working with input from the Engineering group a parametric model was created which enabled investigation into different types of cable based 'bicycle wheel' and hyperbolic paraboloid type roofs. Model initially used as basic CAD input to Finite element GSA model. This was then extended to enable a direct link from Grasshopper to GSA, enabling metrics feedback on the form (such as material usage) compared with the structural performance metrics roof deflections, cable hoop stresses, reactions onto existing structure. With this it was possible to change parameters (such as rim truss depth) and see the update performance metrics in near real-time (3-10 seconds).

Research Value: In-depth technical development and focused design session feedback on parametrically defining frame and tensioned cable based structures. Requirements based on workshops with engineers and defined useful automated interfaces between design software and structural analysis at the concept stage. Findings about the speed of development for interfaces in response to design needs obtained, which typically found in favour of tool and design co-evolution.

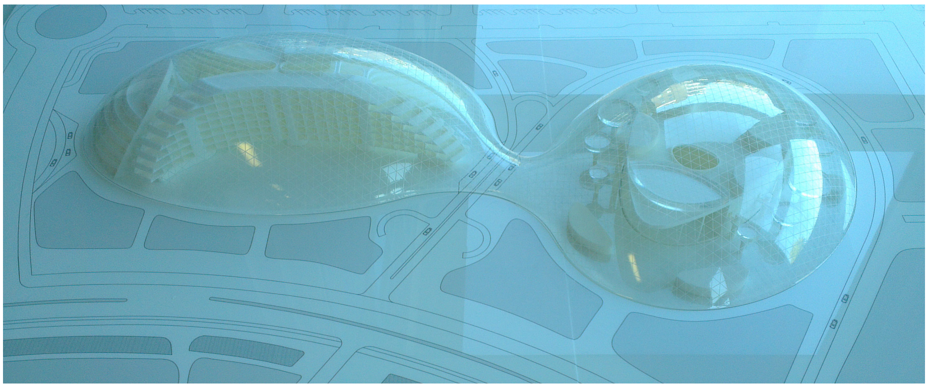


Figure B-10: Physical Model showing 'biomes' covering the conference and combined office and hotel complex. Source Author

B.10 Astana Expo [2255]

- Type: Hotel + Office + Conference Centre
- Location: Astana, Kazakhstan
- Size: Plan-Area 38,000m^2
- Collaborators:
  - Internal: Studio 6, Model shop
- Personal involvement:
  - Stage: Concept
  - Duration: 3 weeks
  - Software: Rhino, Grasshopper, Kangaroo Physics, GSA, F+P Structural Link

Description: Fast-paced development of a hotel and conference centre scheme for the of the 2017 Astana Expo. The concept involved wrapping all the program in a structurally separate series of 'biomes'. These domes where intended to create a climate around the buildings in which was environmentally decoupled from the local extreme temperature ranges by providing an extra insulating zone and containing the buildings exhaust outputs

Involvement: Creation of a plan based dynamic relaxation approach which used a mixture of vertical and inflation forces on various types of grids to enclose the desired volume in an efficient way. Form was then feedback into linear and non-linear buckling analysis for rough satisficing checks as well as production of 3D printed and vac-formed models for visual analysis.

Research Value: Data on developing parametric systems to coordinate definition of engineering and design models simultaneously. Findings both technical and based on designer feedback on the issues of using form-finding methods to create architectural forms. Issues regarding the quick appraisal of thin shell structures especially when considering buckling. Highlighting need to allow architects to understand issues in a general way they can respond to and act on. Study of use comparing full engineer involvement vs basic analysis showing faster development but leading to some unresolved issues later.

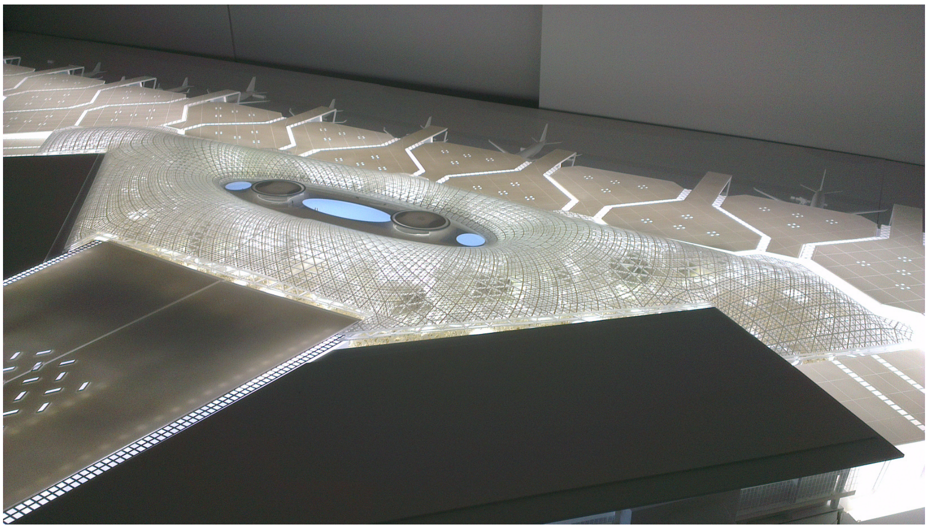


Figure B-11: Model of Doha Airport inner courtyard roof. Source Author.

B.11 Doha Airport[2169]

- Type: Airport + Hotel + Retail
- Location: Doha, Qatar
- Size: Roof Area: 3,300m^2
- Collaborators:
  - Internal: Studio 3, Model Shop
- Personal involvement:
  - Stage: Concept
  - Duration: 2 weeks
  - Software: Grasshopper, SMART Form, GSA, F+P Structural Link

Description: A study to look at enclosing a space created between the existing and proposed extension of Doha airport. This space roughly 350x150m was to have a garden and retail space with an elliptical hotel at its centre. The client requested a covered option to be explored. An efficient steel shell structure was proposed with a shading profile that responds to the different requirements for light of the uses of the space underneath.

Involvement: Working with others in ARD generated a roof option that closely matches the approach taken for the Great Court of the British Museum. Form then was structurally assessed to test for basic structural conformity. A system was developed that used progressive sub-division of the elements in response to a designated floor usage plan. The thickness of the panels frames was optimised to conform to the desired ratio between transparent and opaque.

Research Value: Identification of basic requirements for fast design production of high performance roofs with minimal engineering checks. Investigation of use of form finding optimisation to minimise structural problems under restrictive design constraints. Methods explored to meaningfully compare limited numbers of options' relative structural performance in approximate high-level terms (e.g. steel tonnage).

Figure B-12: Exterior view of Cleveland Clinic. Source Foster + Partners.

B.12 Cleveland Clinic Roof [2192]

- Type: Hospital + Teaching + Research Centre
- Location: Cleveland, U.S.A.
- Size: Roof Size: 17,500m^2
- Collaborators:
  - Internal: Engineering 1 (Structures)
- Personal involvement:
  - Stage: Scheme
  - Duration: 6 Weeks
  - Software: Grasshopper, SMART Form, GSA, F+P Structural Link

Description: Extension of Cleveland Clinic with a teaching hospital wing. A simple square courtyard 168m by 100m was to be covered by a roof structure to create an internal space. Working with the engineering group from inception a steel structure was developed. Initially exploring steel grid-shell like structures but also extending to the exploration of 'Gaussian Vaults' as inspired by Eladio Dieste, before settling on more conventional truss like geometry. Project involved fast iteration of many different forms, as well as significant design optimisation required to keep within a modest budget.

Involvement: Initially design development of parametric models for option exploration. Then progressed to significant engineering analysis of interaction between geometric properties and structural behaviour. Performance and parameter space mapping was undertaken, involving the automatic generation of between 700-4000 individual models with structural performance and cost metrics recorded. This data was then processed by a custom made web-visualisation interface. Allowing for the display of Pareto-optimal fronts, as well as analysis on significance of input parameters to output performance metrics.

Research Value: Full implementation of performance driven design concepts. Integrated geometric and structural model developed from parametric model, with feedback from both engineers and architects on issues of speed and flexibility of option creation. Application and appraisal of automated methods to brute force explore limited but multidimensional design spaces demonstrating practical generation times when considering overnight running. Techniques developed to explore at a performance level design options, including identifying effective interactive data visualisations for this kind of mixed design and performance data. Feedback gained from team members on their understanding visualisation methods.

Figure B-13: Ariel visualisation of the main terminal building roof scape. Source Foster + Partners.

B.13 Mexico Airport[2223]

- Type: Airport
- Location: Mexico City, Mexico
- Size: 470,000m^2,
- Collaborators:
  - Internal: Studio 6, Engineering 1+2 (Structural + Environmental), Visualisations, Model Shop
  - External: NACO (Airport Planners), Geometrica (Grid-shell fabricators)
- Personal involvement:
  - Stage: Competition + Concept
  - Duration: 4 Months
  - Software: Rhino, Grasshopper, Weaverbird (mesh-tools), GSA, F+P structural tools

Description: Development of a new hub airport for Mexico City. Airport proposed to be made of one giant roof structure approximately 600m wide by 1400m long, with a column grid of up to 100m square with a proposed central dome of 170m, making it the largest structure the company has produced and one of the largest structures in the world. The roof is designed to be super light weight made of a modular computer fabricated node and tubular system by Geometrica. This 'super-roof' or 'skin' covers all of the functions of the airport including; drop-off, check-in, customs, flight gates, shopping, baggage handling, arrivals. The roof's scale required it too be as efficient as physically possible.

Involvement: Brought on by the Engineering group to aid in the integration between structural-engineering and the geometric work done in part by my team ARD and also the design studio. Firstly simply to convert geometric models to meaningful structural models in GSA. This then extended to fully investigating the principal form-finding approach along with other members in the ARD group, including comparing form-finding results from Grasshopper with those from GSA's relaxation routines. These where compared both for geometrical differences but also for structural performance. Further work extended to generating an interactive column location tool to visualise the column position effects on spans, program interference as well as a fast generation of the relaxed roof form, all to speed up decision making of the column position with all the stakeholders. After a fixed structural from was derived then the environmental and visual impact of different transparency schemes where investigated.

Research Value: This major project allowed many of the key performance design concepts to be trailed and tested to extreme. For example methods to pass geometry into structural analysis was required to transfer over 250,000 elements. This exercise showed that many of the tools are sufficient to scale however in some instances redevelopment was required. The project demonstrated the need for all parties to be clearly informed of the performance repercussions of their design decisions as these where cases which not enough was known and this impacted progress by involving design rework. The concept of modularity was also tested here with a distributed team needing to undertake different tasks and the concept of a singular owner of the geometry and data process being shown to be insufficient here.

Figure B-14: Visualisation of UAE Pavilion. Source Foster + Partners.

B.14 UAE Expo Pavilion 2015 [2182]

- Type: National Pavilion
- Location: Milan, Italy
- Size: Building Area 3,400m^2, Plan Panel Length 1,000m, Typical panel height 12m
- Collaborators:
  - Internal: Studio 5
  - External: Can Build (GRC Fabricators)
- Personal involvement:
  - Stage: Scheme + Detail + Construction
  - Duration: 5 Months

– Software: Processing, Grasshopper, Galapagos (G.A.), Octopus (G.A.), Python Scripting

Description: A commission to design the expo pavilion for the United Arab Emirates in the 2015 world expo. The concept was to create decorative walls that would enclose the building which was an exhibition space along with national themed restaurant and offices. The walls where intended to be evocative of sand dunes.

Involvement: Along with others in the ARD group initialised the use of a cellular automata diffusion reaction model kernel to imitate the sand deposition of sand-dunes and sand-ripples to be applied to the surface of the walls. Worked to apply optimisation to rationalise the initial free-form centrelines of the walls into arcs and lines, so that during panellisation the panels could compromise of panel arc families. Optimisation acting to minimise unique number of panels whilst maximising panel family distribution. Multi-objective optimisation applied to set individual panel patterns taken from a set of patterns, this was to minimise pattern repetition and number of unique panel arc panel moulds required. Bespoke system developed to automate the mass production of geometry from parametric models. System enables a parametric model to be run saving output geometry and performance figures from different parameter settings and inputs. The system is able to run multiple machines in parallel with out limit task managing each resulting in a scalable infrastructure for mass parametric computation.

Research Value: Significant investigation into various post rationalisation strategies, with heavy application of optimisation routines including genetic algorithms and multi objective approaches. Useful feedback from deign team relating to decision making based on a mixture of aesthetic goals and performance values. Attempts made and insight gained form these attempts including methods to capture previous design versions and options in a repository to aid future progress. Major efforts undertaken in the parallel processing of parametric models with heavy development on a partially automated task manger system to coordinate multiple computers working to work collaboratively.

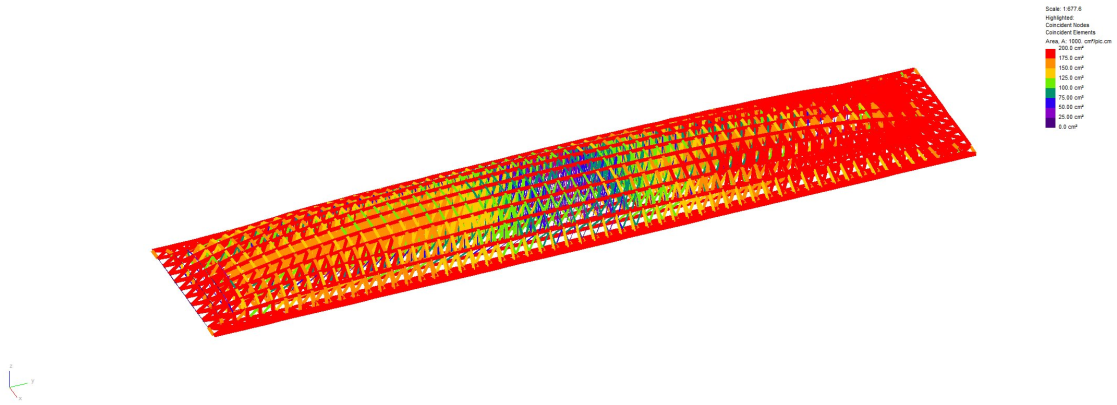


Figure B-15: Space frame after optimisation, colour-scale is indicative of cross-sectional area

B.15 Stadium in Paris [2065]

- Type:
- Location:
- Size: 77,500m^2
- Collaborators:
  - Internal: Engineering 1, Studio 1
- Personal involvement:
  - Stage: Competition
  - Duration: 2 weeks
  - Software: Rhino, Grasshopper, Millipede (Structural Solver/Optimiser), GSA, F+P Structural Tools

Description: Early stage concept development of a large stadium for Paris. Concept was to cover the large stadium with two equally large sliding roof elements measuring 240m by 55m each. These 'sliding doors' would be supported only on the short edges where they would be positioned on sliding bearings enabling them to close together. There was a desire from the design team instigated by engineering that if the roof could be made of a space frame of very fine elements that by light diffraction would not create shadows on the pitch.

Involvement: This problem was posed as an open question to help create a self supporting roof effectively acted as a beam, with as minimal section sizes as possible. An initial process was undertaken to generate a pillow like shape that satisfied the requirements that the roof would be flat on the underside and have no depth on the edges. This was derived analytically with the height derived by a function optimised by the material requirement and a maximum deflection. After generating an form, a principal stresses approach was explored to investigate its potential for creating a bundled tube like design. However this was replaced with a section sizing optimisation embedded in the F+P structural tools to provide minimal sections for given deflections. The resultant sections where found to be an order of magnitude to thick and as such the design was developed in other directions.

Research Value: This short project in close collaboration with the head of structural design provided support and highlighted the benefits and failings of a parametric design system. Showing that in instances of designed flexibility this sped up design process by providing live feedback. However in cases of unexpected changes this took longer to realise. This feedback was acute in case of section size optimisation where developing custom methodologies was time consuming due to it being implemented in the base code and not being explicitly open to edit.

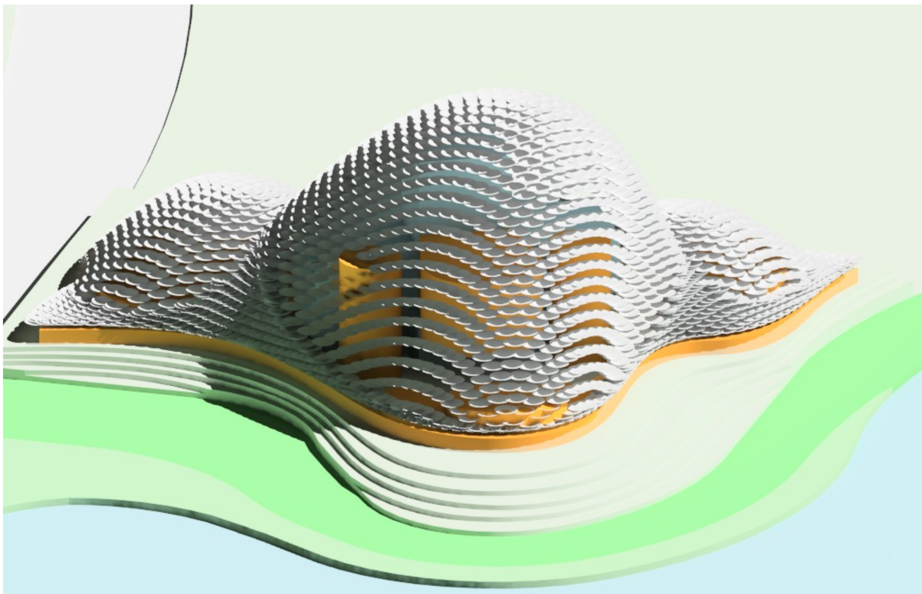


Figure B-16: Busan Opera House external skin with rain water collecting dishes. Source Foster + Partners.

B.16 Busan Opera House [2110]

- Type: Opera House
- Location: Busan, Korea
- Size: Site Area: 20,000m^2
- Collaborators:
  - Internal: Studio 5, Engineering 2 (Environmental)
- Personal involvement:
  - Stage: Competition
  - Duration: 3 weeks
  - Software: Rhino, Grasshopper

Description: Development of the covering skin of the Busan Opera House. This was a relatively quick study to produce a covering which encapsulated the three main programmatic elements in one complete form.

Involvement: Development of many varied concepts for the scheme, including wrapping the volumes in a simulated fabric then remapping the resultant surface for meshing. As well as developing a shading and rain collecting strategy comprising of large disks arrayed on the surface. Developed a program to generate and test for coverage a collecting system.

Research Value: Study gathered data on the application of computational approaches in more architecture only problems. Demonstrating the needs of geometric studies in this domain helping to contrast it against the primarily structural design issues faced elsewhere. Also later feedback was gained about the post rationalisation of the form for engineering proposes which happened after initial concept design.

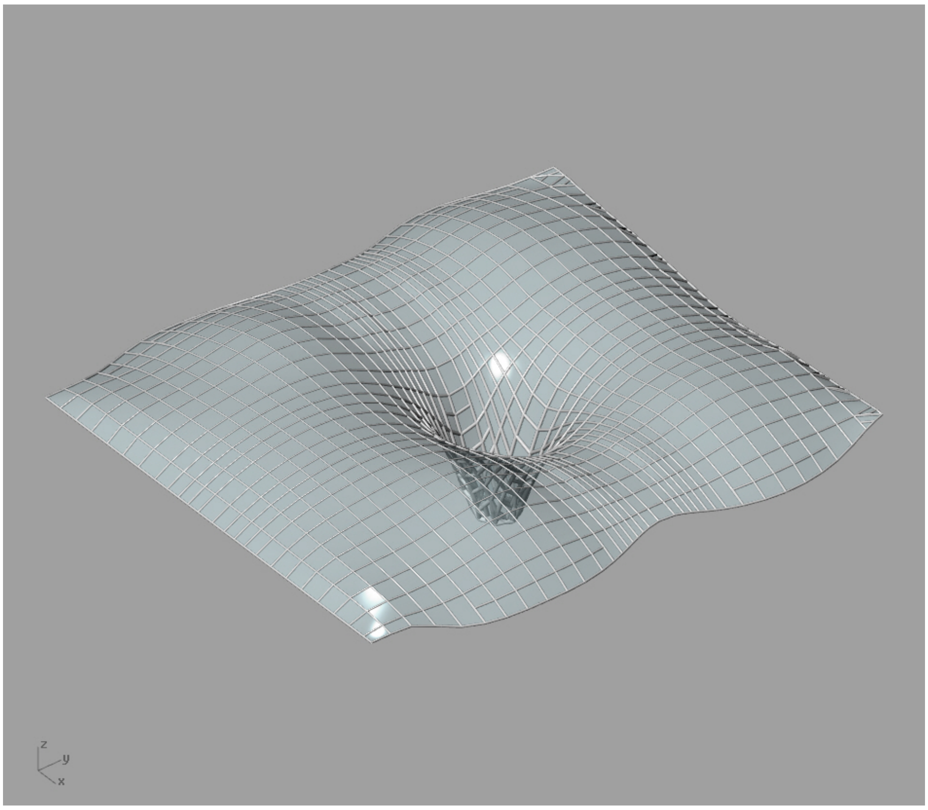


Figure B-17: Overview of repetitive column-roof module

B.17 Xiamen Cruise Terminal [2117]

- Type: Transport
- Location: Xiamen, China
- Size: Roof Module 6000m^2
- Collaborators:
  - Internal: Studio 1
- Personal involvement:
  - Stage: Scheme
  - Duration: 2 days
  - Software: Rhino Grasshopper, Kangaroo Physics

Description: Development of a large module that acted as both column and roof. Form was generated for review and then later rejected in favour of a more traditional form.

Involvement: Basic dynamic relaxation undertaken to generate form, with subsequent refinement to satisfy aesthetic requirements.

Research Value: Investigation of the interaction between architectural and engineering requirements. In this case the ability to modify a purely engineering based solution into one which also works architecturally. Study and feedback highlighted how this is difficult to do with the performance metrics taking significant time to generate.

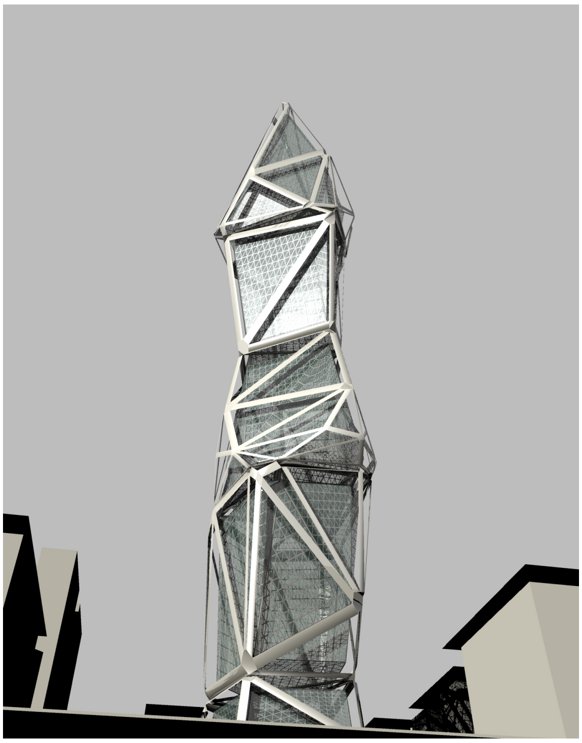


Figure B-18: Example of Voronoi based tower design

B.18 Gemdale Tower [2158]

- Type: Residential Tower
- Location: China
- Size: Plan area 200-400m^2 Height: 100m
- Collaborators:
  - Internal: Studio 6
- Personal involvement:
  - Stage: Concept
  - Duration: 3 days
  - Software: Grasshopper, Mesh tools

Description: Fast option exploration of an extremely thin tower.

Involvement: Development of two options relating to a volumetric voronoi to determine tower shape and sub-module proportion, this was linked to optimisation for building constraints including height and volume. Further tension supported options where also explored.

Research Value: Data gathered about tower typology investigations finding the metrics that where important in deriving meaningful performance metrics in this scenario.

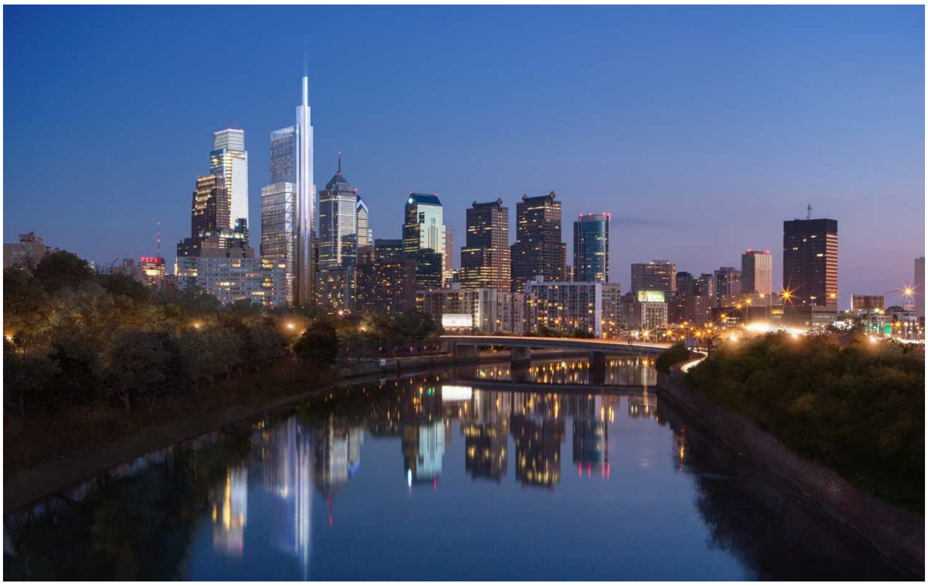


Figure B-19: Visualisation of Project Liberty shown in context. Source Foster + Partners.

B.19 Project Liberty [2161]

- Type: Mixed-use + Office Tower
- Location: Philadelphia, U.S.A.
- Size: GFA: 167,000m^2 NIA: 141,000m^2 , height 341m
- Collaborators:
  - Internal: Studio 1
- Personal involvement:
  - Stage: Scheme
  - Duration: 1 week
  - Software: Rhino, Grasshopper

Description: Option exploration of enclosing spaces at the top of two large towers to create sky gardens.

Involvement: Generation of Geodesic dome like ellipsoids by subdivision of initial surface. Also creation of relaxed and basic modularised unit systems for the enclosure.

Research Value: Demonstration of the issues involved in applying pure engineering solutions to architecturally complex schemes. Identified issues regarding how to adapt these and that control is required to allow them to be adaptive to the design context.

Figure B-20: Overview of Tocumen project next to existing airport. Source Foster + Partners.

B.20 Tocumen Airport [2034]

- Type: Airport
- Location: Tocumen, Panama
- Size: Floor Area: 80,000m^2
- Collaborators:
  - Internal: Engineering 1 (Structures)
- Personal involvement:
  - Stage: Detail
  - Duration: 2 weeks
  - Software: C Sharp Scripting, F+P tools, Etabs, web

Description: An addition to the existing airport, comprising of a large new terminal linked to the existing building. The linking structure was visually and geometrically unsophisticated structure, however it had complex loading due to it being both the elevated passenger connection above and a top-hung baggage handling frame below requiring a high portal frame structure, without bracing. This was in a highly seismic area and was proving very difficult to tune whilst keeping weight down.

Involvement: I developed an automatic procedure to generate static and dynamic analysis data from models. There where three main structural sections, and each was able to utilise 9 different section sizes. All possible combinations of these where generated and analysed within ETABS the structural software used by the engineering team for this project. This brute-force-search not only provided a number of working examples but also was able to show which sections had what effects on the different Eigen modes of the dynamic structure, with this information a suitable option could be chosen.

Research Value: Thorough investigation and parallel development of large scale deigns and performance space exploration. Research highlighted the required batch processing capabilities to automatically run data production and manipulate parametric model use. Additionally investigations in data visualisation for understanding the link between design and performance space. This project clearly demonstrated the benefits of a computational approach over more manual methods.